NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

**THE SMALL THEATER LEVEL MODEL:
AN EXTENSION OF FTLM**

by

Thomas A. Nelson

September 1994

Thesis Advisor: Samuel Parry

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)

2. REPORT DATE
September 1994

3. REPORT TYPE AND DATES COVERED
Master's Thesis

4. TITLE AND SUBTITLE
THE SMALL THEATER LEVEL MODEL: AN EXTENSION OF FTLM (U)

5. FUNDING NUMBERS

6. AUTHOR(S)
Nelson, Thomas A.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE
A

13. ABSTRACT (Maximum 200 words)

This study outlines the design, implementation, and testing of the Small Theater Level Model (STLM). The purpose of this research was the first in a sequence of efforts to determine if the course of action perception methodologies of the Future Theater Level model (FTLM) could be used in the small theater. Currently, there are no other models that have the capability to provide the small theater commander with perceptions of the enemy's intent. Additional modifications were made to FTLM in order to more accurately portray small theater operations: the addition of a range dependent attrition algorithm, high resolution of aviation assets conducting sensor observations, and the ability to provide a dynamically employable reserve force into the battle. Testing of the model was based on the development of a scenario similar to battles fought at the U.S. Army's National Training Center (NTC). Multiple replications were run, using different sensor performance standards, to evaluate the model's ability to convert reconnaissance into perceptions of the enemy's intent, and the use of a deterministic attrition algorithm in a stochastic model. A discussion of the results concludes with the requirement to conduct further testing of the course of action perception methodology, design more elaborate tactical rule sets for the employment of the reconnaissance assets and the reserve force, and ultimately, develop a more rigorous scenario that can be compared to actual NTC results.

14. SUBJECT TERMS
Stochastic, Small-Theater, STLM, Combat-Model, Bayesian

15. NUMBER OF PAGES
114

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18
298-102

Approved of public release; distribution unlimited

THE SMALL THEATER LEVEL MODEL:
AN EXTENSION OF FTLM

by

Thomas A. Nelson
Captain, United States Army
B.S., United States Military Academy, 1984

Accesion For	
NTIS CRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and / or Special
A-1	

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1994

Author: ______________________
Thomas A. Nelson

Approved By: ______________________
Samuel Parry, Thesis Advisor

____
LTC George Prueitt, Second Reader

____
Peter Purdue, Chairman
Department of Operations Research

ABSTRACT

This study outlines the design, implementation, and testing of the Small Theater Level Model (STLM). The purpose of this research was the first in a sequence of efforts to determine if the course of action perception methodologies of the Future Theater Level model (FTLM) could be used in the small theater. Currently, there are no other models that have the capability to provide the small theater commander with perceptions of the enemy's intent. Additional modifications were made to FTLM in order to more accurately portray small theater operations: the addition of a range dependent attrition algorithm, high resolution of aviation assets conducting sensor observations, and the ability to provide a dynamically employable reserve force into the battle. Testing of the model was based on the development of a scenario similar to battles fought at the U.S. Army's National Training Center (NTC). Multiple replications were run, using different sensor performance standards, to evaluate the model's ability to convert reconnaissance into perceptions of the enemy's intent, and the use of a deterministic attrition algorithm in a stochastic model. A discussion of the results concludes with the requirement to conduct further testing of the course of action perception methodology, design more elaborate tactical rule sets for the employment of the reconnaissance assets and the reserve force, and ultimately, develop a more rigorous scenario that can be compared to actual NTC results.

EXECUTIVE SUMMARY

This research was initiated in response to the lack of ability in current models to provide the small theater commander with a tool that recognizes and models the benefits of intelligence on the battlefield. The small theater is defined as those operations that occur at, and below, the brigade level. The Future Theater Level Model (FTLM), a relatively new, untested, large theater model, has demonstrated progress in the ability to model intelligence and provide estimates of opposing force intentions. The overall goal of this thesis was to take FTLM and modify it to perform these same functions in the small theater: assessing placement of units and employment of reconnaissance assets. This research is the first of a sequence of efforts to produce a mature model.

To pursue the small theater level model (STLM), it must be determined if it is capable of overcoming the current limitations of high resolution models: specifically, large overhead support requirements, in terms of people, equipment, and time. The architecture and attributes in the original design of FTLM make it an excellent candidate model for the small theater. In addition to the short set-up and execution time, the model provides analysts with the capability to rapidly examine alternative operational concepts and force mixes under conditions of uncertainty.

The specific objectives of this thesis were to take FTLM and modify it to create STLM, by incorporating the course of action (COA) perception methodology, modifying the maneuver and attrition modules, and evaluating the capability of the new model using a scenario developed from the U.S. Army's National Training Center. The requirement to modify the maneuver module was based on the need to amplify certain capabilities such as the projection of reconnaissance assets and the commitment of a reserve force. A range

dependent attrition algorithm was added to the model to provide greater resolution given the difference in theater size between FTLM and STLM.

The resulting analysis was an evaluation of the model changes applicable to the small theater, determining what the model was capable of providing to an analyst, and providing direction for further research. Using a helicopter as a reconnaissance platform, three different sensor performance standards were applied against three different opposing force courses of action. Thirty simulation runs for each combination of sensor capability and opposing force course of action were evaluated to determine if the course of action perception methodology could be used in the small theater. Analyzed separately, but in conjunction with these simulation runs was the ability of the helicopter to receive missions, proceed to an observation post, and conduct reconnaissance. The accuracy of this reconnaissance was expected to be a function of the sensor performance standard.

The final model analysis was the evaluation of the range dependent attrition algorithm and the inclusion of the reserve force in the model. These results were also drawn from the output of the simulation runs. Using simple rules to employ the reserve force, the model was evaluated to determine if the reserve force deployed properly in support of the scenario objectives. Unit attrition was examined for sensible outcomes and to determine if it provided the resolution necessary for continued use in the small theater model.

The study results and analysis indicated several issues and areas that warrant further research. First, the course of action methodology did not meet the required expectations. Additional research is needed to provide a better insight and understanding into the perception updating methodology. Second, the model input parameters must be examined for their impact on various outputs and later inclusion in the mature model. Third, tactical decision rule sets that are more realistic for the employment of the reconnaissance and the reserve force must be developed. The final area is the development of a realistic scenario

that can be evaluated against historical data such as that found at the NTC. As STLM evolves to maturity, it should be capable of assisting the analyst with many critical issues:

- Employment of reconnaissance assets on the battlefield.
- Implications of various sensor capabilities as they relate to intelligence.
- Locations of critical terrain where reconnaissance provides the greatest insight into the enemy intent.
- Employment of ground assets to maximize survivability, including the use of a reserve force in support and counterattack missions.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

Simulation modeling provides decision makers with many insights into military problems. Although modeling is not new to the military, the advances in computer hardware and software have enabled modelers to design simulations that can handle ever greater computations and tasks. Models are typically designed to answer, or address, a particular question, or range of issues. As such, models have many characteristics that help to classify their purpose and capabilities. The common link among most models is the limitation of deterministic processes.

Recently, new technologies have led to the introduction of smaller models with reduced operating requirements, that can provide the same insights into complex problems. The Future Theater Level Model (FTLM) was designed for the Conventional Forces Analysis Division, J-8, The Joint Staff, to model the inherent uncertainties in theater-level combat using stochastic inputs and processes and overcome the limitations of current models: [Ref. 1: p. 1]]

- Deterministic design.
- Failure to use intelligence and perception within the model.
- Linear orientation of the battlefield.
- Large number of experienced personnel to operate.
- Intense preparation and execution time.

The original design version of FTLM started as a research effort to examine variability and uncertainty in an aggregated theater representation for force structure analysis. What separates FTLM from other models is its ability to calculate course of action perceptions based on detections and sensor observations. It is also unique in that it can be setup, run

and operated by one person, on a typical IBM compatible computer, using Microsoft Windows™.

A. BACKGROUND

Of particular interest in this research are those models that assist the small theater commander or decision maker with brigade and smaller unit issues. Currently, these models or simulations are limited to high resolution models and suffer the same limitations of the larger theater level models. There are two models currently in use that warrant comparison with FTLM as a small theater model. The Combat Arms Task Force Engagement Model (CASTFOREM) and JANUS(A) are both high resolution models with stochastic inputs and processes.

1. CASTFOREM

Implemented in 1983, CASTFOREM is used to model brigade and below combined arms conflicts for weapon systems and tactics evaluation. It is primarily intended to model intense battalion-level battles, up to one hour in length. It is considered very high resolution for conventional and directed energy weapon systems, with resolution to the item level. Processes are modeled probabilistically using Monte Carlo techniques. The simulation is used for combined arms ground conflicts and includes support helicopters, fixed-wing aircraft, air defense systems and dismounted fire teams. The simulation is capable of modeling conventional warfare with limited chemical and nuclear effects. Directed energy weapons, including lasers and high-energy microwave systems are also modeled. CASTFOREM is extremely flexible and can accommodate any terrain, using digitized terrain data, or weapon systems for which data are available. Weather and ambient light conditions are constant throughout the battle, while battlefield obscurants, smoke, and dust are modeled as dynamic clouds. [Ref. 2:p. D-3]

The US Army Training and Doctrine Command (TRADOC) considers the model useful as an analysis tool and seminar driver for equipping forces, fighting unfamiliar forces, battle command, and C4I. CASTFOREM takes several days to setup and run, as well as a full team of experienced people to operate.

2. JANUS(A)

Implemented in 1988, JANUS(A) is used for combat development analysis and training. The model undertakes analytical studies of both current and new doctrines related to strategy, policy and weapon system development. As a training tool, its primary mission is to train battalion level, and below, personnel in battle-focused training to enable junior leaders to synchronize the battlefield. JANUS(A)'s secondary mission is to function as a seminar exercise driver for the tactical commander's development program. [Ref. 2: p. D-13].

TRADOC has identified JANUS(A) as an analysis tool, seminar driver and exercise driver for equipping forces, fighting unfamiliar forces, night fighting, battle command, C4I, and continuous operations. Like CASTFOREM, JANUS(A) can take several days to setup and run. The number of experienced operators required to run JANUS(A) is limited by the mission for which JANUS(A) is used. While it is capable of being run as a stand alone unit by one person, JANUS(A) is maximized as a tool when an entire staff is brought in for a command post exercise.

3. FTLM

The architecture and attributes in the original design of FTLM make it an excellent candidate model for the small theater. In addition to the short set-up and execution time, the model provides analysts with the capability to rapidly examine alternative operational concepts and force mixes under conditions of uncertainty. [Ref. 1: p. 2] It is a closed form simulation that uses existing attrition models and permits nonlinear battle. FTLM

uses a network for ground, air and logistics maneuver. All data for the model are input into an ASCII text file as part of an initialization file.

The ground network is established as part of the initialization files. Nodes, referred to as physical nodes, are usually associated with cities, road junctions, changes in terrain, or places where units engage. Arcs, or transit nodes, are also part of the initialization files, and take on characteristics of the terrain (including cover, concealment and defensive positions). Units move along these nodes by designated corridors and specified courses of action, given an objective for that course of action. Unit composition, dictated by the operator, is the basis for a unit's lethality, movement rate, and capabilities.

The air network is a grid specified by an overall size and an interior grid of squares. Aircraft move from an air base to grid centers and the midpoint of any grid edge. Aircraft can be either fixed or rotary wing and move to an objective by calculating a route that is a minimization algorithm of the shortest distance and least resistance to perceived enemy air defenses. The aircraft mission is specified by the operator and can be any combination of 17 different types of missions (including reconnaissance, close air support and air interdiction). Aircraft are susceptible to detection, jamming, and enemy air defense systems.

FTLM was designed to use detections and sensor observations to evaluate possible opponent courses of action, while CASTFOREM and JANUS(A) use detections primarily to generate target lists for engagements and attrition.

B. RESEARCH OBJECTIVES

This thesis will take previous work by Schmidt, *Design Methodology For FTLM*, [Ref. 3] and Johnson, *Quantifying The Value of Reconnaissance Using Lanchesterian Type Equations*, [Ref. 4] and extend it to the small theater. By making several

modifications to the original FTLM model, the new small theater model can be used as an aid to the commander in assessing placement of units and employment of reconnaissance assets. This research is the first of a sequence of efforts to produce a mature model.

The objective of this thesis is to take the course of action (COA) perception methodology in the Future Theater Level Model (FTLM), and apply it to a small theater-battalion sized force (hereafter referred to as the Small Theater Level Model-STLM). It must be determined if it is capable of overcoming the current limitations of high resolution models. The goal of the resulting analysis is to provide a quantitative measure of the benefit of intelligence: determination of an enemy's true course of action and what are the best sensors to use to gain that information using STLM.

C. PROBLEM STATEMENT

This thesis will answer the following questions regarding the suitability of FTLM to the small theater:

- Does the COA perception methodology provide the expected results in the small theater? Specifically, can the COA perception updating correctly determine the ground truth of an opposing side's course of action?

- Do the modifications to the air maneuver model allow representation of individual aircraft to operate as sensor platforms and conduct sensor observations?

- Can the aircraft follow simple tactical decision rule sets to carry out those observations?

- Do the sensor observations of small sized elements correctly translate into the calculation of the expected unit combinations, given sensors of varying performance standards?

- Does the scaling of parameters to reflect smaller unit sizes produce any unexpected problems that cannot be corrected?

- **Does the modification to the ground maneuver model provide a side the opportunity to correctly employ a reserve unit?**

- **Can the reserve unit commit itself to the battle by following simple tactical decision rule sets?**

- **Does a deterministic attrition algorithm produce unexpected problems with the other stochastic aspects of FTLM in the small theater?**

- **What is required, in terms of additional modifications and further research, to the present model to produce a mature STLM?**

D. SCOPE, LIMITATIONS AND ASSUMPTIONS

This research is unique in that there are currently no other models that provide the COA perception methodology to the small theater. Previous COA perception methodology work has been limited to large theaters of operation. While all of the initial assumptions for FTLM remain valid, the most critical assumption for this research is that the model can be scaled down to a small theater of operations. Given that the modifications made to FTLM are credible, the remainder of this thesis will be devoted to developing a network based NTC scenario, scaling parameters to reflect the smaller unit sizes, and analyzing the model operation and output to answer the critical research questions.

The limitations of this research are the fact that the parent model, FTLM, is unverified and not validated. Also, this study will use fictional data: specifically, force compositions, attrition data, movement rates, aircraft survivability, rates of fire, and priority allocations of fires. Further limitations include the use of entry level tactical decision rule sets as a basis for initial determination of suitability in the model, the use of a deterministic attrition algorithm, and the inability to collaborate the outputs of STLM with another similarly verified model.

E. OUTLINE SUMMARY

Chapter II discusses the framework and primary uses of FTLM. Because of the large amount of historical data from the National Training Center, it was chosen as the only available test-bed for the small theater level model. The archive data provides a solid foundation for examining causal relationships between a unit's ability to detect the enemy and its performance in defeating that enemy.

In considering sensors available in the small theater, the one that provides this level of force commander with the greatest reach is aviation. By detecting and identifying the enemy early, and more importantly, determining his true course of action, the small theater commander can use his limited indirect fire assets to attrite the enemy before the close fight. Chapter II details the changes made to the original FTLM, including the use of helicopters, to develop the small theater level model. Because FTLM, and certainly STLM, are not fully mature models at this time, this chapter will also describe the desired characteristics of a mature STLM.

Chapter III describes the scenario used to evaluate STLM, which was drawn from historical battles at the U.S. Army's National Training Center. While no two battles fought there are the same, there is a common thread in the task force defensive battle. In Johnson's research, he concluded that proper use of reconnaissance was a definite multiplier on the battlefield. [Ref 4: p. 8] While the defensive battle is not the only time that reconnaissance is needed, it is a good starting point for evaluating a model's ability to provide the small theater commander with enemy courses of action perception.

After initial testing of the model design, Chapter IV details the analysis of the model output. Answers to initial questions such as:

- Does the model accurately represent the small theater combat process?
- Does the model provide insight not already found in other models?

- Does the model represent systems well enough to define good measures of effectiveness?

At this point, the strengths and weaknesses of the model should become evident, as well as identification of areas requiring further refinement and recommendations for future changes. Chapter V contains the conclusions of this research and provides specific topics for future study.

II. SMALL THEATER METHODOLOGY

A. GENERAL

The methodology and techniques used to convert FTLM into a small theater model are a combination of programming modifications and changes to the scenario initialization file. The initialization file changes are primarily scaling of values, unit sizes and assets, to more accurately represent the small theater. Unit representation in FTLM is aggregated at the brigade and division level. This research will make the changes necessary for STLM to represent battalions and companies. Appendix A is a detailed listing of the required entries for the initialization file. A sample initialization file is in Appendix B.

Many of the algorithms in the original FTLM were placeholders, and they were not considered critical in the initial evaluation of FTLM. Some modules were either omitted, or simplified until approved modules could be agreed upon. One of the critical aspects in the design of FTLM was the requirement for object oriented programming [Ref. 5]. This had several benefits:

- Easy swapping of modules to facilitate the needs of the analyst.
- Modular changes reduce programming costs and save time.
- Increased efficiency both in computation and speed of the overall model.

These concepts were followed in all modifications and refinements that led to the evolution of STLM.

B. FUNCTIONAL AREAS

To document the changes made to FTLM, it is best to examine them as they are outlined in the FTLM Summary of Model Concept. [Ref. 1] This will provide a more

cohesive understanding of how STLM relates to its predecessor. Figure 1 is a schematic layout of the FTLM architecture. [Ref. 1: p. 5] For the purposes and scope of this study, the logistics module was not incorporated into STLM. This chapter is focused on the three remaining critical areas necessary for STLM: Command, Control, Communications and Intelligence (C3I), maneuver, and attrition.

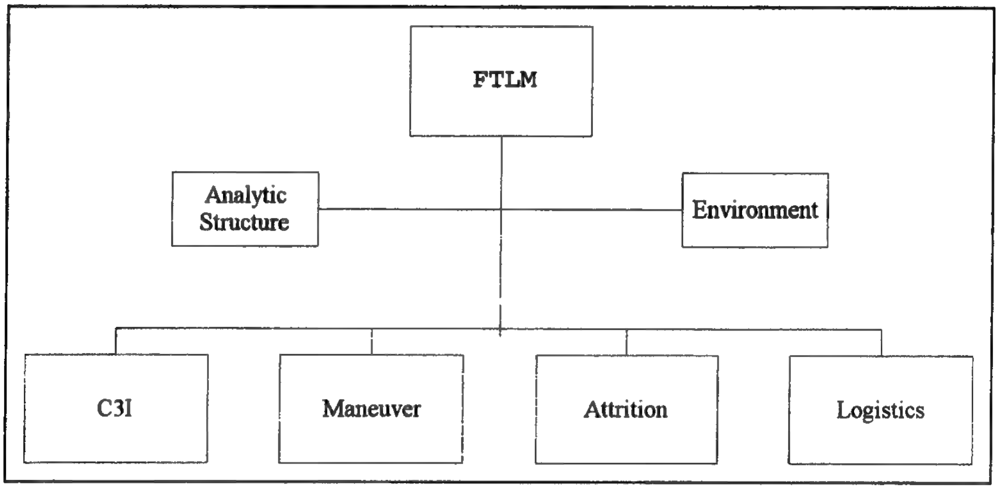

Figure 1. FTLM Top Level Architecture

As in FTLM, the small theater C3I encompasses determining when, where and how much combat power the enemy can bring to bear. Maneuver is the movement of forces to gain positional advantage on the battlefield, [Ref. 6: p. 2-13] and attrition is the result of the application of combat power at a given place and time.

1. C3I

It is important to understand that the C3I process is the central focus of FTLM [Ref. 1: p. 9] and remains unchanged for STLM. The C3I process is divided into two, independent, collections of events:

- Detections at physical nodes.
- Sensor observations from reconnaissance assets.

What has been altered to more accurately portray the small theater scenarios used in this thesis are those events that update the COA perception. The generation of detections can be thought of as coming from some form of continuous observation platform, such as a satellite. The observations from reconnaissance assets are periodic and directed by tactical decision rule sets. [Ref. 1: p. 20] Throughout this thesis, the reader must understand the distinction between detections and sensor observations as they are described above.

a. COA Update from Unit Detection

When a unit reaches a physical node, a detection is triggered which in turn is used in the COA perception update. Detection is a function of the unit's transit time across the previous arc, and the detection rate associated with that arc. This detection rate is part of the model set up, and it is specified in the initialization file for each transit node. Let

$$\bar{\lambda}_i(k) = \sum_{j \in J(k)} \left\{ \left[\lambda_j(k)\right]\left[e(k)_{ji}\right]\left[U_{ji}(k)\right] \right\}, \tag{1}$$

be the mean detection rate on occupied arcs over the entire corridor in a course of action, i, for period k. This is obtained by summing over all arcs, j, the product of an arc's detection rate, λ_j, the non-zero exposure time for arc j, in course of action i, $e(k)_{ji}$, and the unit size weight on arc j, in course of action i, $U_{ji}(k)$. [Ref. 7: p. 10] The probability of detection in [(kΔ, (k+1)Δ], given a specific course of action, C_i, can then be defined as

$$P\{D(k) = d(k) | C_i\} = e^{-\bar{\lambda}_i(k)\Delta} \left[\frac{\left(\bar{\lambda}_i(k)\Delta\right)^{d(k)}}{d(k)!} \right]. \tag{2}$$

From this probability, the model then uses Bayesian updating to calculate the COA perception probability. [Ref. 7: p. 11] While this update is suitable in the lower resolution

of FTLM, it is inappropriate for the higher resolution STLM. The small theater commander does not have access to these continuous type sensors.

To correct for this, the detection rate was adjusted in the initialization files so that the updating occurred, but it would, in effect, be separated from the COA perception processes. The detections themselves are necessary to generate periodic reconnaissance requirements, so the code could not be removed in entirety. Also, this method provides the analyst flexibility in choosing detection rate values. As a result, the weight that these detections have on the COA update was reduced by decreasing the detection rate for each transit node. It must be emphasized that the detections are still necessary to trigger reconnaissance missions; but the detections do not impact on the COA perception update.

To determine suitable values for the detection rate, all sensors were removed from the model. The result was a COA perception update based solely on the detection rate and the update cycle. The detection rate for each transit node was decreased by magnitudes of ten until there was no change in the COA perception at the end of a simulation replication. Table 1 shows the perception of the ground truth COA at the end of a replication given the different detection rate values.

TABLE 1. DETECTION RATE TUNING

Detection Rate	COA Perception
10	1.000
1	0.845
0.1	0.562
0.01	0.467
0.001	0.333

For example, if the detection rate was 0.1, the COA perception probability of ground truth was approximately 0.562 at the end of the replication. When the value was

decreased to 0.001, all COAs were equally likely when the run was complete. This was the desired result, and the detection rate for transit nodes was set to 0.001 for the run design. STLM uses the inverse detection rate in the initialization files, therefore the actual entry value was 1000.

b. COA Update from Sensor Observations

To calculate a side's course of action from sensor observations, the model uses asset-counting sensors given that a unit or side has been detected in $[(k\Delta, (k+1)\Delta]$, as described in paragraph a, above. The sensor takes observations that report assets of type j at node n. These counts are assumed to conform to a normal distribution with mean, [Ref. 7: p. 13]

$$m_{nj}(k) = \frac{\displaystyle\sum_{l=1}^{b_n} \frac{s_l(n,j;k)}{\tau_{nj}^2(l)}}{\displaystyle\sum_{l=1}^{b_n} \frac{1}{\tau_{nj}^2(l)}}\,; \tag{3}$$

and variance, [Ref. 7: p. 13]

$$v_{nj}^2(k) = \frac{1}{\displaystyle\sum_{l=1}^{b_n} \frac{1}{\tau_{nj}^2(l)}} \tag{4}$$

where $s_l(n,j;k)$ is the lth sensor observations at node n, of asset type j, during period k, $\tau_{nj}^2(l)$ is the variance of the error of the lth sensor observation counting asset type j, at node n, and b_n is the number of observations for node n during the sensor period. The number of sensor observations taken on a node will determine how well the mean and variance reflect the ground truth.

Once the model has computed the mean and variance of the sensor observation in (3) and (4) above, the mean and variance are computed for all active nodes during period $[(k\Delta, (k+1)\Delta]$, in corridor, α, where the calculated mean is [Ref. 7: p. 13]

$$\overline{m}_{\cdot j}(\alpha;k) = \sum_{n \in \overline{N}(\alpha:k)} m_{n_{ij}}(k); \tag{5}$$

and the variance is [Ref. 7: p. 14]

$$\overline{v}_{\cdot j}^{\,2}(\alpha;k) = \sum_{n \in \overline{N}(\alpha,k)} v_{nj}^2(k). \tag{6}$$

For each corridor, α, at time k, and given the total assets of type j, on course of action, c, there is a normal density function, $\xi(\alpha;\mu,\sigma^2)$, [Ref. 7: p. 14], computed by:

$$\xi_j(\alpha;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}(\sigma_j^2(\alpha) + \overline{v}_j^{\,2}(\alpha;k)^{1/2}} \exp\left\{\frac{-1/2\,(\overline{m}_{\cdot j}(\alpha;k) - \mu_j(\alpha,k;c))}{(\overline{v}_j^{\,2}(\alpha;k) + \sigma_j^2(\alpha,k;c))}\right\}^2; \tag{7}$$

where $\mu_j(\alpha,k;c)$ is the mean from the unit's authorized Tables of Organization and Equipment, for the given corridor and course of action. In the model, the variance, $\sigma_j^2(\alpha,k;c)$, is 10% of the mean. This computed value is the posterior probability, $\Pi(c;k)$, at time $k\Delta$, that a side is following course of action c. To obtain the posterior probability in the next sensor update, $[(k\Delta, (k+1)\Delta]$, [Ref. 7: p. 14]

$$\Pi(c;k+1) = \frac{\Pi(c;k)\prod_{\alpha}\prod_{j}\xi_j(\alpha;\mu_j(\alpha,k;c),\sigma_j^2(\alpha,k;c))}{\sum_{c}[\Pi(c;k)\prod_{\alpha}\prod_{j}\xi_j(\alpha;\mu_j(\alpha,k;c),\sigma_j^2(\alpha,k;c))]}; \tag{8}$$

If no observation is taken, then [Ref. 7: p. 14]

$$\xi_j(\alpha,k;c)=1. \quad (9)$$

If no other detections occur on other corridors, the prior probability, $\Pi(c;k)$, is unchanged and the posterior distribution is identical to the prior belief. Once a detection occurs on another corridor, the new prior becomes $\Pi(c;k+1)$. Accordingly, the sum of the probabilities of all perceived courses of action is one. [Ref. 7: p. 14]

The previous section has very briefly highlighted the course of action perception algorithm using Bayesian updating techniques. A more complete discussion of this is provided in Schmidt's thesis, *"Design Methodology for FTLM."* [Ref 3: p..57-60] This is the single most important aspect of FTLM that separates it from all other models. No other model, or simulation provides the user or analyst with enemy course of action perceptions and updating. As discussed in Chapter I, other models use detections primarily to generate target lists.

2. Maneuver

In FTLM and STLM, the maneuver model is defined as the interaction between units and their environment. [Ref. 1: p. 31] In STLM, there are two components to the maneuver model: ground maneuver and air maneuver. Again, the logistics aspect was not considered in this research.

a. Ground

Recall that units move along an arc-node path, and physical nodes are used to represent objectives, defensive positions, bases, targets and connections between arcs. [Ref. 1: p. 32] Transit Nodes (arcs) connect physical nodes and have attributes that identify homogeneous terrain conditions and trafficability for units. The network is further divided into collections of physical and transit nodes, called corridors. The association of units to specific corridors constitutes a course of action. Units travel from a base or starting point, in accordance with a course of action, towards an objective. More than one

corridor can be used for any given course of action. The actual route traveled is determined using a shortest distance algorithm from the base to the objective.

Because the model does not limit movement to one side, any number of scenarios can be evaluated, including offensive and defensive actions, and movement to contact. Both sides have COAs that are linked in the initialization files. For example, one side's COA has a counter response COA by the other side. This research was limited to the evaluation of one side defending a position, with the other side attacking. The entire ground network, including corridors and courses of action is specified by the analyst in the initialization files.

The availability of a reserve unit is critical to the small theater commander. STLM has the capability to specify a reserve unit and initiate its movement to a support or counterattack position. The rules governing movement of the reserve unit are:

- Only after direct fire engagement has begun.
- Movement to the physical node in support of the perceived enemy COA.

These rules, which were nonexistent in FTLM, were designed only for this version until more elaborate rule sets could be devised.

For example, after units have come into direct fire range, the model determines which perceived enemy COA has the highest probability. The reserve unit then moves to the physical node that supports the counter COA.

b. Air

The air network is a two dimensional grid, subdivided into squares, over the ground network. Aircraft are assigned to a squadron which is located at an air base on a ground physical node. All aircraft start and end their missions at an air base. Missions are generated when an opposing unit is detected entering a physical node as described in the

paragraph above. There are 17 types of missions for aircraft in the model and are they listed in Appendix A, paragraph 19.u.

For this study, the number, type, and aircraft missions was limited to two reconnaissance helicopters. To make the model more realistic, changes were made that allow helicopters to orbit a specific location for a predetermined amount of time. This feature is not available in FTLM.

In STLM, aircraft operate autonomously or in a sortie as defined in the initialization files. Once a mission is generated, aircraft take off and proceed to an observation post (OP) where they conduct observations on physical and transit nodes. The number of missions can be limited by restricting the number of detections that can generate a mission at a physical node. The number of observations taken can be limited by the length of the orbit time and the time between updates to the COA perception cycle. After completing orbit, helicopters check the mission queue for another mission. If another mission is in the queue, then the helicopter proceeds to the new OP. These simple tactical decision rules can be expanded for later versions of STLM. All aircraft can be subject to counter-air, ground attrition, jamming and enemy air defense. Because of the objectives of this thesis, these features were suppressed.

3. Ground Attrition

The ground attrition module is a major programming change for STLM. In FTLM, attrition only occurs when two or more units occupy the same physical or transit node. In the small theater, this is inadequate because of small unit sizes and the direct fire range between opposing units (weapons systems) can span more than one node. The Bonder range dependency attrition algorithm was adopted for STLM. Even though this attrition method is deterministic, it was chosen as a preliminary means of attrition until a more sophisticated, stochastic method could be decided on.

As units move through physical and transit nodes, the model compares distances to determine if any two opposing weapon systems are within direct fire range. For example, let A_{ij}^0 be the maximum attrition rate, and A_{ij} be the range dependent attrition rate for weapon system i against the opposing side's asset j. At each time step in the model, the algorithm determines if any i, j combinations are within the maximum range, R_{max}, for each respective weapon system i, and target j. If so, the model applies an attrition algorithm using the Bonder range dependency equation, [Ref. 8: p. 88]

$$A_{ij} = A_{ij}^0 \left(1 - \frac{R}{R_{\max[i]}}\right)^{\mu} . \tag{10}$$

Accordingly, the attrition rate for each weapon system is a function of the rate of fire from the initialization files, a percentage of the current range to target, its maximum range, and the Bonder scaling parameter, μ. This equation can be used for both direct and indirect weapon systems by setting the Bonder scaling parameter to zero for indirect fire weapons, and a value greater than zero for direct fire weapons. The closer the value of the scaling parameter is to zero, the smaller the range dependency effect. When the value is one, the attrition rate decreases linearly to zero at the maximum range. For values greater than one, the attrition rate decreases as a convex function. Figure 2 illustrates the effects of the Bonder scaling parameter on attrition. [Ref. 8: p. 88] In this example, the maximum attrition rate, A^0, is 0.85, the maximum range, R_{max}, is 3000 meters, and the three scaling parameters are 0.5, 1.0, and 2.0.

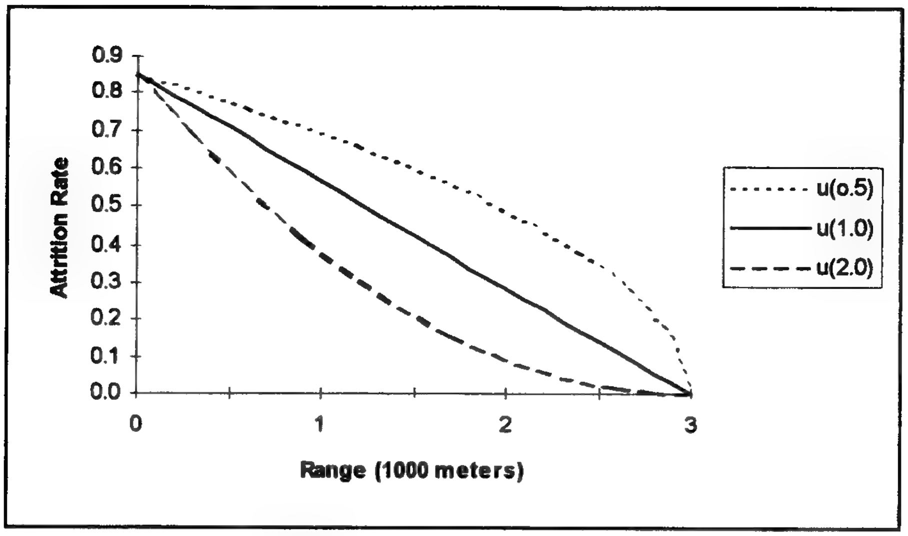

Figure 2. Effects of Different Scaling Parameters

III. SCENARIO/RUN DESIGN

A. SCENARIO

After reviewing the Army Research Institute's files of NTC battles, a defensive scenario was chosen to conduct the analysis for three reasons:

- There are at least ten task force (battalion) defensive battles recorded over the same terrain from which to design the network for the scenario.
- Johnson's previous work [Ref. 4] described the impact that reconnaissance had on attrition using a task force defensive scenario similar to the one chosen for this study.
- The terrain over which these battles have been fought offers multiple identifiable courses of action for both forces.

The NTC task force defensive mission directs the commander to conduct a deliberate defense and deny the enemy, a regimental size force, penetration of that terrain. In the battles that were examined, each commander used his reconnaissance assets differently; some successful and some not as successful. From the playbacks, it was evident that the most difficult aspect of intelligence gathering was correctly determining, early in the battle, the location of the enemy attack. In most cases, a good fix on the enemy was not obtained until the close fight was imminent and the advantage of employing early indirect fires was not realized for the task force commander.

One of the unique aspects of the NTC playbacks is the ability to examine unit traces: paths of where units started, their movement, and where they ended when the battle was over. This provided the traces of ground truth enemy courses of action and defensive unit actions. Figure 3 illustrates how a force might view the NTC terrain and divide it into possible corridors.

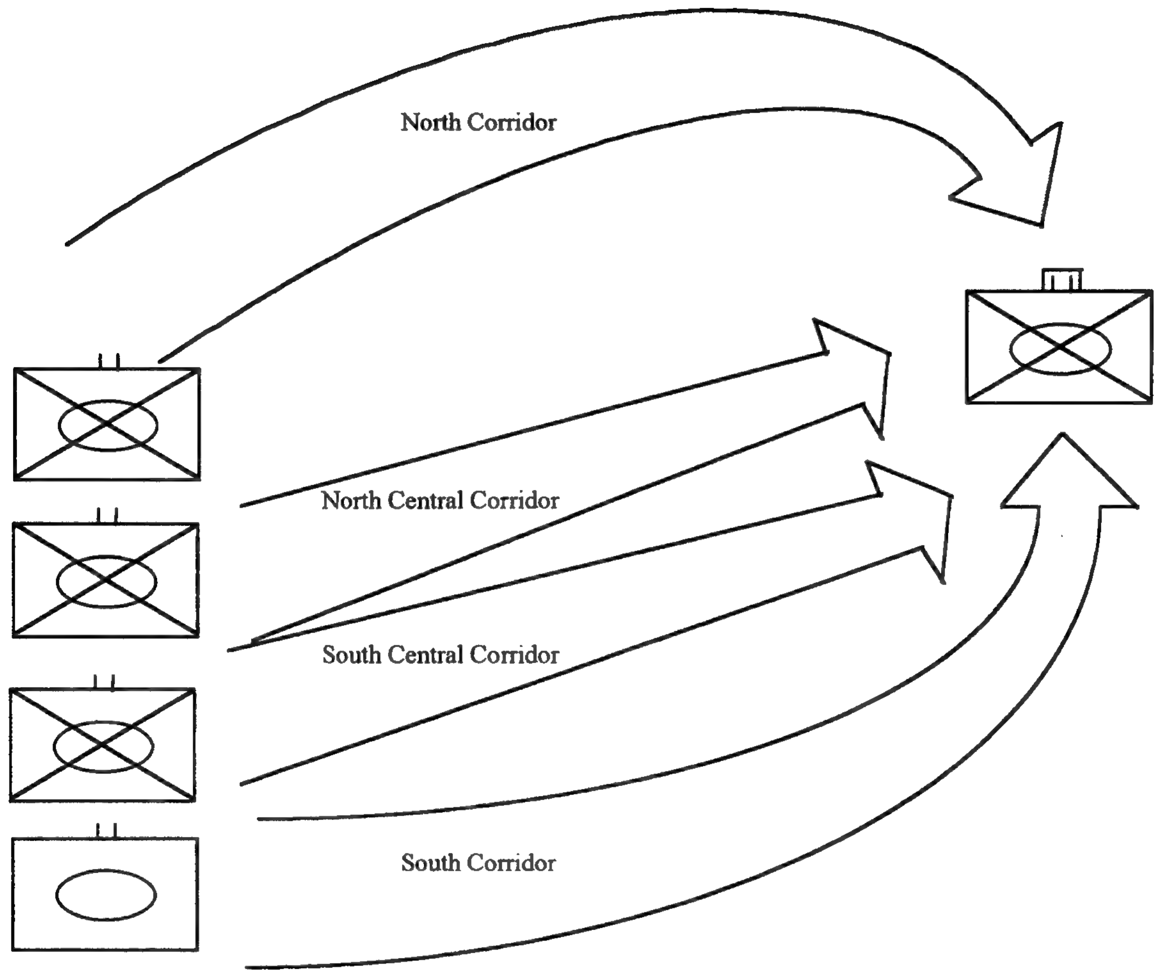

Figure 3. Battle Graphics

Enemy forces, approximately regimental strength, attack from west to east along four corridors, while the friendly forces attempt to deny penetration and destroy the enemy.

With the aid of the playbacks and military maps, a network was developed which captured the general routes and positioning of forces throughout the battle. This network

is show in Figure 4. The attributes for each node, as well as the distances, type terrain and width for each arc in the network are listed in Appendix C.

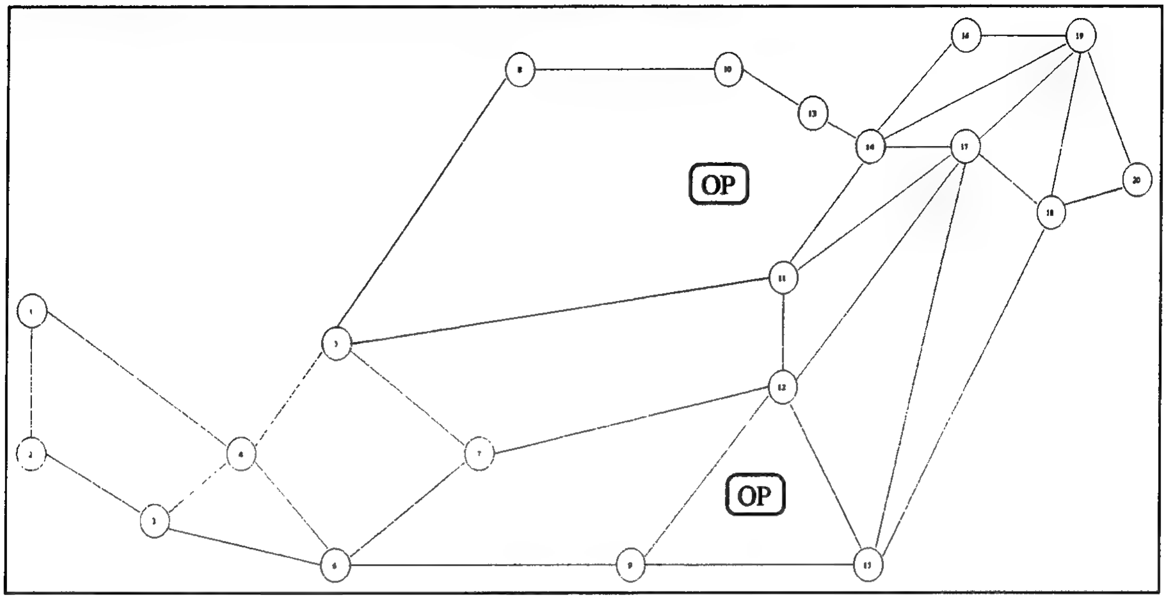

Figure 4. Network Representation of NTC Terrain

Friendly forces, hereafter referred to as Blue Forces, have assumed defensive positions on physical nodes 14, 17, and 18, with a reserve force on physical node 19. Each of these nodes contains a company sized force. The enemy, Red Forces, start on nodes one and two; with the tank and artillery battalions on node one, and the three mechanized battalions on node two. The Red force objective is node 17. The routes that the Red forces take to the objective are specified in the ground truth course of action.

The network is subdivided into corridors, shown in Figures 5 through 8. The Red force can attack along any of the four corridors, in any combination of units. When units are assigned to corridors, then a specific combination of units and corridors constitutes a course of action. Three courses of action for the Red force were designed to evaluate STLM, and they will be detailed, subsequently.

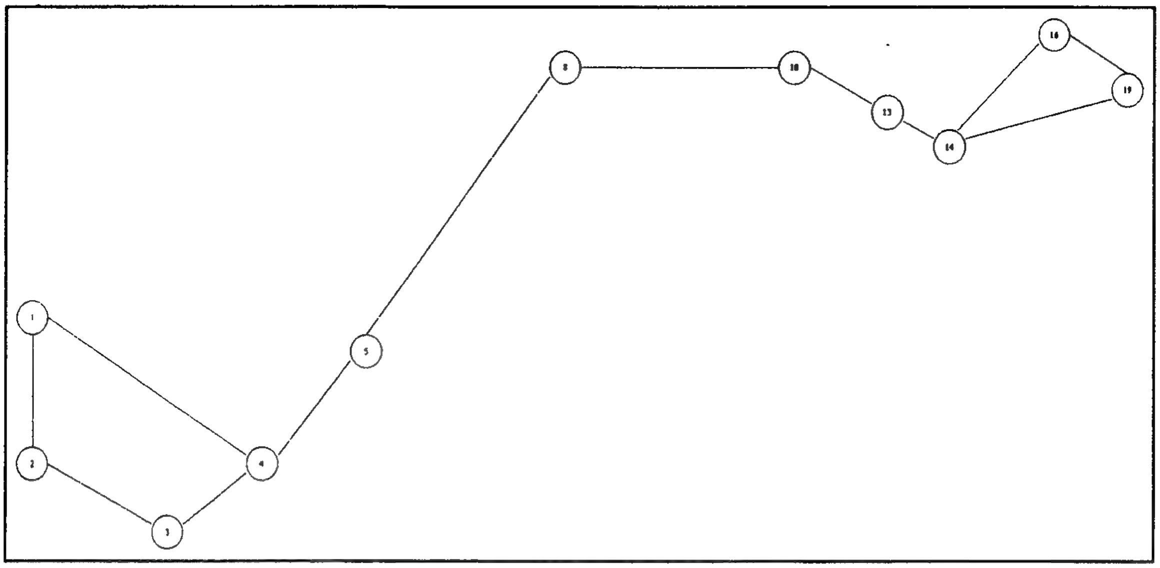

Figure 5. North Corridor

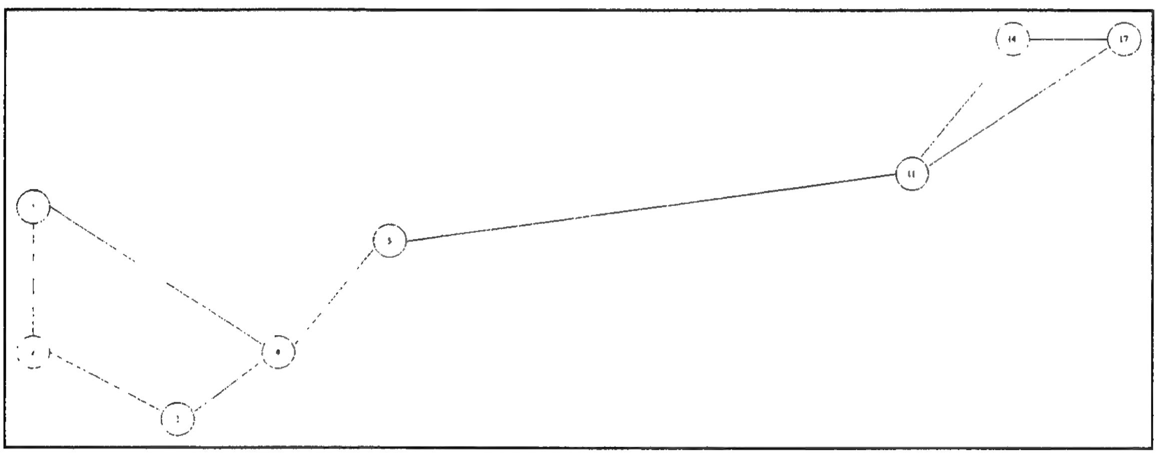

Figure 6. North Central Corridor

When the Red forces are within direct fire range, the Blue reserve force moves forward to support the defense on the corridor which is associated with the highest probability for Blue's perception of Red's COA. For the three scenarios used in this study, this supporting position will be directly linked to the corridor the Blue force perceives the Red tank battalion is using. The purpose of linking the commitment of the Blue reserve company to the engagement of the Red tank battalion was to evaluate the

attrition impact that the additional Blue assets would have on what is perceived to be the greatest enemy threat, hence, the outcome of the battle.

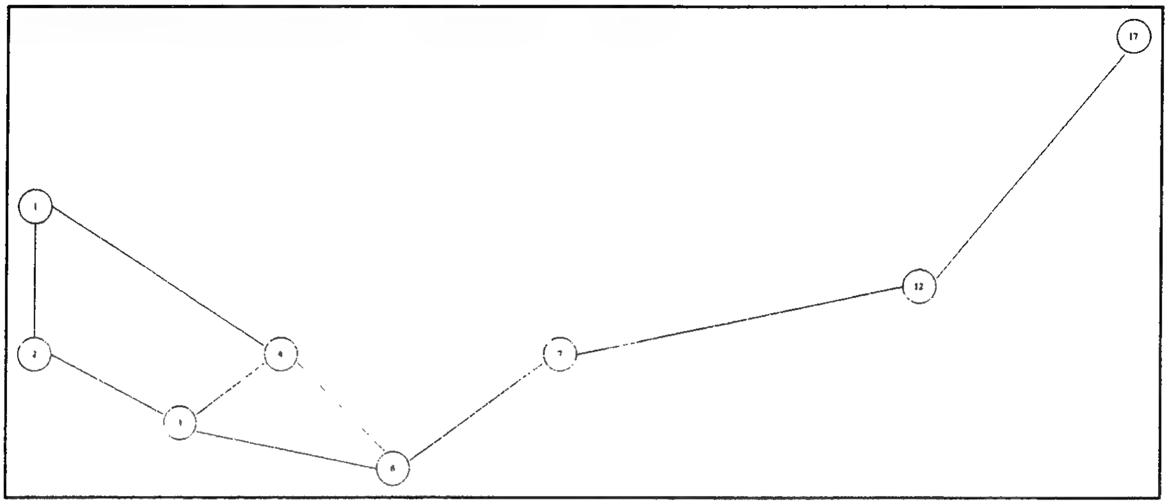

Figure 7. South Central Corridor

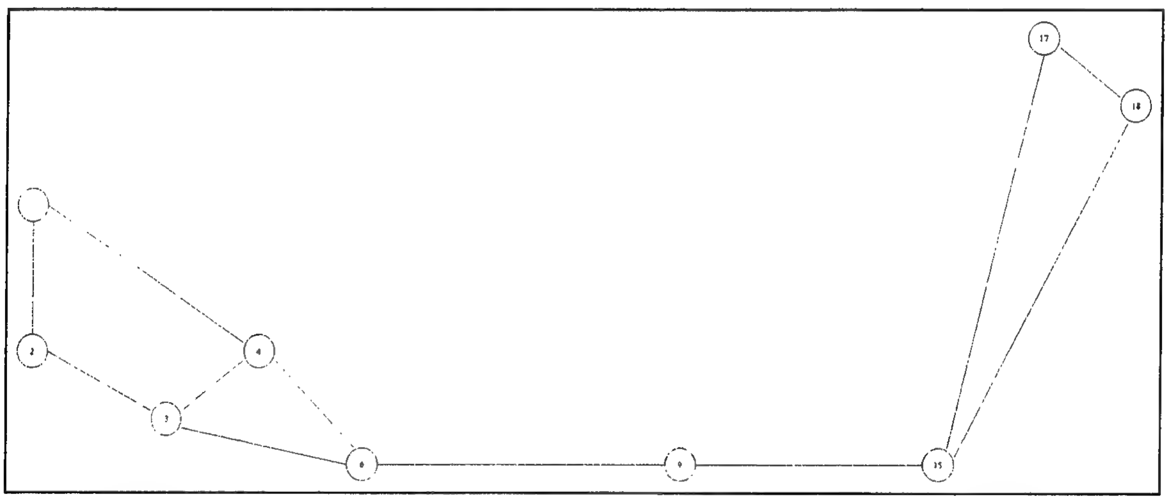

Figure 8. South Corridor

For example, when the Red force is within direct fire range and the course of action perception with the highest probability links the Red armor battalion to the South Corridor, then the Blue reserve company will move forward to physical node 18. If the Blue perception of Red's course of action is incorrect, then the Blue reserve company will move to support the wrong position.

1. COA 1

The first course of action entails all Red forces, including the tank battalion, taking the North corridor. The purpose of this design was to evaluate the relative quickness of the Blue reconnaissance assets to locate the Red Force and determine the ground truth course of action.

2. COAs 2 and 3

COA 2 splits the Red force along three corridors: the tank battalion on the North Central Corridor, two motorized rifle battalions on the South Corridor and one motorized rifle battalion and one artillery battalion on the South Central Corridor.

COA 3 splits the Red force on three corridors: the tank and artillery battalions on the South Central Corridor, two motorized rifle battalions on the South Corridor, and one motorized rifle battalion on the North Central Corridor.

These two courses of action were intentionally 'nearly identical', with the only difference being the corridor linked to the tank battalion. The purpose of these two COAs was to evaluate the ability of the sensor to observe and correctly identify the unit combinations on the respective corridors and make a determination of the correct enemy course of action. The artillery battalion was linked to the South Central Corridor in both COA 2 and 3 to prevent the sensor from determining the ground truth based solely on identifying the artillery.

1. Unit Structure and Equipment

Units are composed of atoms, the smallest pure type asset structure in STLM. For example, a tank atom is a collection of pure tanks. Different types of atoms can be organized together to form a unit, or combined arms team. The collection of units forms a side. Outlined below are the structure and true asset counts for both the Blue and Red forces.

a. Blue Forces

The Blue task force is composed of two armor companies and two mechanized companies. The task organization of the company teams are two balanced teams, one mechanized heavy team and one tank heavy team for the reserve. The internal combat power of the task force is M1 tanks and M2 Bradleys. The number of company assets was chosen to simulate an evaluation of a light platoon, where the number of platoon assets is three M1s or M2s, instead of the conventional four M1s or M2s. Assets supporting the task force, but not internal to the organization, include one direct support artillery battery and a pair of reconnaissance/attack helicopters. The primary mission of the helicopters is to provide forward reconnaissance to the commander. They are equipped with sensors that provide intelligence on the threat's assets. The quantities and types of equipment are shown in Table 2.

TABLE 2. BLUE EQUIPMENT AND QUANTITIES

Equipment	Nomenclature	Quantity
Tanks	M1	24
Infantry Fighting Vehicles	M2	24
Artillery (tubes)	M109	8
Reconnaissance Helicopters	RAH-66	2

b. Red Forces

The enemy is organized and equipped similar to a Motorized Rifle Regiment (MRR) of a Motorized Rifle Division. The MRR has three Motorized Rifle Battalions and one Tank Battalion. For this study, it is assumed that the MRR does not task organize; each company of each battalion maintains unit integrity. Also, the MRR is equipped with

artillery and air defense assets that are internal to the organization. The equipment used by the Red force in these scenarios is listed in Table 3.

TABLE 3. RED EQUIPMENT AND QUANTITIES

Equipment	Nomenclature	Quantity
Tanks	T72	78
Infantry Fighting Vehicles	BMP	90
Artillery (tubes)	2S1	24
Air Defense	ZSU 23-4	4

2. Sensors

For the purposes of this research, only the Blue Force has sensors, and they are in the form of rotary wing aircraft. When a detection is triggered by a Red unit entering a physical node, a helicopter is dispatched to an observation post. When it arrives, it immediately conducts sensor observations (with individual standard deviations for detecting Red assets). The locations of the observation posts are specified in the initialization file. In this scenario, there are two observation posts, one between the North and North Central Corridors, and the other between the South and South Central Corridors as illustrated in Figure 4.

B. RUN DESIGN

The model was run to evaluate the suitability of STLM in the small theater. Given the programming changes to the maneuver model, the scenarios were designed to evaluate the ability of the Blue force to determine the true Red force course of action (ground truth) using different sensors. Table 4 shows the run matrix, with replications, for the COAs and sensors. Three different sensor packages were used to evaluate the course of action

perception update for each of the three courses of action. This resulted in nine different runs with thirty replications for each run. Additionally, the suitability of adopting the Bonder attrition algorithm was evaluated for inclusion in the model by examing the output files for expected results.

TABLE 4. RUN MATRIX

	Sensor		
	Alpha	Bravo	Charlie
COA 1	30	30	30
COA 2	30	30	30
COA 3	30	30	30

C. OUTPUT

Each replication can be evaluated by examining the output files. Table 5 lists the filename extension for each type of output file. The filename, designated 'fn' in the table, is the filename of the initialization file. A separate set of files is built for each replication. The replication is identified by the number after the first letter in the filename extension. A description of each of the files is summarized below.

TABLE 5. OUTPUT FILENAMES

Type File	Filename Extension
History	fn.h01
Blue COA	blue.c01
Red COA	red.c01
Air Units	fn.m01
Ground Units	fn.m01
Attrition	fn.a01

1. History File

a. Explanation

The STLM history file is in chronological order and contains all unit actions, both ground and air, all sensor observations, and COA perception updates. Each replication generates a new file. Depending upon the number of sensor observations, the file can be quite extensive in length, often more than 150 pages. The other output files were designed to aid the analyst wanting specific information about a specific area and are explained in below. The sensor observations are only contained in the STLM history file. A sample observation extracted from a history file is explained below.

After a sensor platform (helicopter) arrives at an observation post, it conducts observations on predesignated physical/transit nodes. The accuracy of what it observes is a function of the standard deviation of that sensor to observe a particular asset on a particular node. Once it has observed the node and recorded the asset count for each of the assets it is capable of observing, it computes the probability of that combination of assets belonging to a specific unit combination. It computes the probability for all possible unit combinations. From the last line of the example below, the probability of a unit combination of one tank company, one artillery battery, and two mechanized companies, given that it has observed 18 tanks, 10 artillery pieces, and 27 BMPs, is 0.422931. The columns of asset numbers listed after the probability are the mean number of assets and variance, given that unit combination. The extract begins with a helicopter arriving at an observation post, OP1. Many of the unit combinations had probability zero and were deleted from the extract since the listing of all unit combinations is several pages long.

b. Extract

```
Time   49.09868 BLUE mission package B16 starts sensor observations at OP1
           The following physical/transit nodes are to be observed:
                  NODE.06
                  TRANSIT.11
                  TRANSIT.12
•
•
•
Time   49.09868 BLUE sensor B.SENSOR.2 searching arc TRANSIT.11
           RED combat unit assets
                  RED.TANK count - 18
                  RED.BMP count - 27
                  RED.ARTY count - 10
COMPANY combinations are as follows:    (ARMOR, ARTILLERY, MECHANIZED)
```

Unit Combination	Posterior	Red Tank		Red BMP		Red Arty	
(6,3,6)	0.000000	73.43	(1.38)	71.68	(2.16)	22.50	(1.47)
(6,0,6)	0.000000	73.81	(1.32)	72.05	(2.14)	0.36	(0.85)
(6,2,6)	0.000000	73.56	(1.36)	71.80	(2.15)	15.49	(1.31)
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
(2,3,3)	0.000001	25.73	(0.92)	41.91	(1.66)	22.74	(1.35)
(3,2,1)	0.000001	38.24	(0.95)	15.91	(1.10)	15.62	(1.13)
(1,3,3)	0.000001	13.12	(0.78)	41.96	(1.64)	22.78	(1.33)
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
(1,1,1)	0.008377	13.07	(0.61)	15.79	(1.03)	8.07	(0.82)
(2,0,2)	0.019191	25.80	(0.78)	29.65	(1.37)	0.12	(0.50)
(1,0,2)	0.041069	13.07	(0.61)	29.66	(1.35)	0.09	(0.43)
(2,2,2)	0.092456	25.77	(0.85)	29.63	(1.40)	15.62	(1.13)
(2,1,2)	0.196590	25.79	(0.82)	29.64	(1.39)	8.08	(0.88)
(1,2,2)	0.198100	13.10	(0.70)	29.64	(1.39)	15.63	(1.11)
(1,1,2)	0.422931	13.09	(0.66)	29.65	(1.37)	8.07	(0.85)

2. COA Files

Each COA file lists one side's COA perceptions in chronological order according to the update cycle. The model assumes all courses of action are equally likely until the first sensor observation is recorded and processed in the next cycle update. The time interval for the perception updates is specified in the initialization data. For each replication, two

files are created: one for each sides' perception of the other. For this study, Red's perceptions of Blue were not evaluated since the Red force did not have any sensors. The COA perceptions are also included in the history file. The following extract is part of the output file for Blue's course of action perception of Red when COA 1 is the ground truth.

TIME	CYCLE	R.COA.1	R.COA.2	R.COA.3
1.00	1	0.333333	0.333333	0.333333
2.00	2	0.000000	0.500000	0.500000
3.00	3	0.000001	0.500000	0.500000
4.00	4	0.000001	0.499999	0.499999
5.00	5	0.000001	0.499999	0.499999
6.00	6	0.000001	0.499999	0.499999

3. Air and Ground Files

There is a separate file for both air and ground maneuver that lists the air and ground unit actions: planning, movement, detection and operational status. These files are also in chronological order and provide an excellent source for tracing units during the course of the simulation.

a. Air Units

The following is an extract for one package from the air units file. The file is in chronological order, by side, package or sortie, action taken and the location. A typical listing for an air unit would be to launch, report arrivals at grids in the network, start orbiting, end orbiting, report grid arrivals when exiting the network, and disband at the air base.

TIME	SIDE	PACKAGE	ACTION	LOCATION
75.99	BLUE	B18	LAUNCH	GRID.10
76.49	BLUE	B18	ARRIVES	GRID.10
76.65	BLUE	B18	ARRIVES	GRID.9
76.88	BLUE	B18	ARRIVES	GRID.18
77.11	BLUE	B18	ARRIVES	GRID.27
77.34	BLUE	B18	ARRIVES	GRID.36
77.34	BLUE	B18	START.ORB	GRID.36
78.34	BLUE	B18	END.ORB	GRID.36
78.50	BLUE	B18	ARRIVES	GRID.37
78.73	BLUE	B18	ARRIVES	GRID.28

78.96	BLUE	B18	ARRIVES	GRID.19
79.10	BLUE	B19	LAUNCH	GRID.10
79.19	BLUE	B18	ARRIVES	GRID.10
79.19	BLUE	B18	DISBANDS	GRID.10

b. Ground units

The following is an extract of the ground units history file. All action taken by ground units are recorded by time, side and unit. Units arrive, plan and depart at physical nodes. They can be detected by the opposing side when they enter a physical node. Finally, when they reach a breakpoint in asset strength, they are reported as broken along with the location that they reached at the breakpoint threshold.

TIME	SIDE	UNIT	ACTION	(FROM) LOCATION	(TO) LOCATION
0.00	RED	RED.1	DETECTED	NODE.01	
0.00	RED	RED.1	ARRIVES	NODE.01	
0.00	BLUE	BLUE.AIR.1	DETECTED	NODE.21	
0.00	BLUE	BLUE.BASE	DETECTED	NODE.21	
•	•	•	•	•	•
•	•	•	•	•	•
•	•	•	•	•	•
54.77	RED	RED.3	DEPARTS	NODE.05	NODE.08
54.80	RED	RED.2	DEPARTS	NODE.05	NODE.08
54.80	RED	RED.4	DEPARTS	NODE.05	NODE.08
55.04	RED	RED.5	ARRIVES	NODE.05	
•	•	•	•	•	•
•	•	•	•	•	•
•	•	•	•	•	•
97.00	BLUE	BLUE.5	BREAKS	NODE.19	
99.00	RED	RED.1	ARRIVES	NODE.10	
99.00	RED	RED.1	DETECTED	NODE.10	
99.10	RED	RED.1	PLANNING	NODE.10	
•	•	•	•	•	•
•	•	•	•	•	•
•	•	•	•	•	•
130.77	RED	RED.5	PLANNING	NODE.14	
130.96	RED	RED.5	DEPARTS	NODE.14	NODE.16
144.00	RED	RED.5	ARRIVES	NODE.16	
144.00	RED	RED.5	DETECTED	NODE.16	

4. Attrition File

The attrition file is a chronological listing of asset strengths for both sides. Each side's assets are reported every minute by unit designation. The entries represent the current number of surviving assets. Although perceptions are stochastically determined, the attrition is deterministic at this time.

TIME	SIDE	UNIT	R.TANK	R.BMP	R.ARTY	B.TANK	B.IFV	B.ARTY
1.00	BLUE	BLUE.1				6.00	6.00	0.00
1.00	BLUE	BLUE.2				9.00	3.00	0.00
1.00	BLUE	BLUE.3				6.00	6.00	0.00
1.00	BLUE	BLUE.4				3.00	9.00	0.00
1.00	BLUE	BLUE.5				0.00	0.00	8.00
1.00	RED	RED.1	39.00	0.00	0.00			
1.00	RED	RED.2	13.00	30.00	0.00			
1.00	RED	RED.3	13.00	30.00	0.00			
1.00	RED	RED.4	13.00	30.00	0.00			
1.00	RED	RED.5	0.00	0.00	24.00			
•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•
128.00	BLUE	BLUE.1				0.93	0.33	0.00
128.00	BLUE	BLUE.2				6.08	0.00	0.00
128.00	BLUE	BLUE.3				5.09	2.36	0.00
128.00	BLUE	BLUE.4				0.11	5.20	0.00
128.00	BLUE	BLUE.5				0.00	0.00	3.25
128.00	RED	RED.1	18.80	0.00	0.00			
128.00	RED	RED.2	0.00	25.93	0.00			
128.00	RED	RED.3	0.00	25.60	0.00			
128.00	RED	RED.4	0.00	25.76	0.00			
128.00	RED	RED.5	0.00	0.00	23.28			

IV. MODEL ANALYSIS

The model analysis focuses on four central areas: COA perception, sensor observation sensitivity, attrition, and areas that warrant further research. The COA perception analysis concentrates on the comparison of ground truth to Blue's perception of each of Red's courses of action. In other words, can the model convert observations (asset counts), using the Bayesian update techniques, into an accurate picture of what the Red force is really doing? The next step is to evaluate whether this perception becomes sufficiently clear, and early enough in the battle, such that a commander could take action on the results.

The sensor observation sensitivity analysis is necessary to determine if changes in the sensor variance have the expected results in both the posterior unit combination probability and the COA perception update. As the sensor variance increases, the number of possible unit combinations that are likely should also increase. This should increase the amount of time it takes the model to perceive ground truth or worse case, prevent it from determining ground truth altogether. The real world implications are minimum necessary sensor variances: a highly accurate platform versus one that cannot make accurate asset counts.

The attrition analysis is based upon reasonable expectations. Recalling that the attrition formulation is dependent upon the Bonder range parameter, the expected results should be attrition that increases as units, specifically assets, get closer together.

While the model proved to be consistent in some aspects, STLM is not yet a mature simulation and there are some areas that need attention. Each section details the results of the STLM runs, discusses problems encountered, and highlights what the mature model

should be capable of providing the analyst. The final chapter will recommend areas for further research.

A. SENSORS AND DETECTION

In STLM, sensors can take any form on the ground, or in the air. For this analysis, the sensor platform was a helicopter. The platform observation capabilities are a function of the distribution for each type of asset that it can sense and the node on which it is sensing that asset. These values come from the initialization files and are input by the analyst. Therefore, if the analyst had data or specifications for a particular type of real-world sensor, they could be used in STLM.

In this research, three different sensors were used, each without regard to terrain or other environmental effects. Therefore, the standard deviation for sensing an asset was the same throughout the network. The order of magnitude of the standard deviation for each of the three sensors was arbitrary, but loosely related to unit size.

The first sensor, Sensor Alpha, has a compact performance distribution and was almost deterministic in its capability. For each type of threat asset, the standard deviation of the sensor was less than 0.1. The second sensor, Sensor Bravo, had a standard deviation that was approximately equivalent to a platoon, or one-third of the size of an atom in STLM. An atom is the smallest collection of pure assets in STLM and a collection of atoms forms a unit. Each of the Red Battalion units in the scenario had three atoms. The last sensor, Sensor Charlie, had a standard deviation for each asset that was equivalent to the unit size, or three times the standard deviation of Sensor Bravo.

To show the impact of the varying quality of information that each of the different sensors had on observing assets, three observations were collected from each sensor. From that, the unit combinations that comprised 90% of the posterior probabilities were

tabled for comparison. The expectation would be that the worse the sensor, the more unit combinations it would take to comprise the 90% probabilities. The three sample table of the collected output is in Appendix D. Table 6 is an extract of one sample from each sensor. The table lists the sensor used, the unit combination, the posterior probability of that unit combination, the expected number of each type of asset given that unit combination, and the observed assets count aligned with the correct unit combination.. The double asterisk in the unit combination column is the actual unit combination that was observed.

TABLE 6. SAMPLE OF SENSOR OBSERVATIONS

			Expected			Observed		
Sensor	**Units***	**P(Unit)**	**Tank**	**Arty**	**BMP**	**Tank**	**Arty**	**BMP**
Alpha								
	(1,3,2)**	1.00	14	22	30	14	22	30
Bravo								
	(3,0,5)	0.06	39	0	78			
	(2,0,6)	0.07	26	0	89			
	(3,1,6)	0.15	39	8	89			
	(3,0,6)**	0.67	39	0	89	37	0	86
Charlie								
	(5,1,6)	0.05	65	8	92			
	(5,0,6)	0.06	65	0	92			
	(4,2,6)	0.06	52	16	92			
	(3,1,6)	0.11	39	8	92			
	(3,0,6)**	0.13	39	0	92	61	7	110
	(4,1,6)	0.23	52	8	92			
	(4,0,6)	0.28	52	0	92			

*Unit combinations are (Armor, Artillery, Mechanized)
**Actual unit combination

Using Sensor Alpha, only one unit combination was necessary and the posterior probability was always greater than 99.9%. The difference between the observed and expected asset count was less than three assets When Sensor Bravo was used, the number of unit combinations to comprise the 90% sample was between two and four. The

range of the maximum posterior probability for the most likely unit combination was between 0.65 and 0.79. As expected, as the standard deviation of the sensor increased, so did the number of unit combinations to complete the 90% sample. Even with the increased standard deviation, the sensor was still able to postulate the correct unit combination. The last sensor, Sensor Charlie, required between eight and twenty unit combinations to comprise the 90% sample and the range of the maximum posterior probability was between 0.13 and 0.27. With this sensor, the most probable unit combination was not the actual unit combination observed in two of the three samples.

This section has illustrated the impact of the sensor standard deviation on the posterior probability of the unit combination. Both FTLM and STLM use these results to derive and update the COA perception probabilities. Whereas FTLM also uses the detection routine to update the COA perception, the sensor observations (asset counts) are the only intelligence available for STLM to compute the COA perception probabilities.

B. COURSE OF ACTION PERCEPTION

The COA perception is the centerpiece of both FTLM and STLM. The ability to model the uncertainty of combat has far reaching implications. Deterministic models fail to provide the commander with course of action perception. Without this piece of critical information, commanders and analysts must rely on military judgment to draw conclusions about the enemy's intent. Not only does each commander synthesize information differently, but it is possible that two commanders can reach completely different conclusions with the same information. When the stochastic processes used in STLM have matured, STLM will be capable of providing analysts with a method of fusing information and providing probabilistic conclusions on enemy intent. From a different perspective, a commander might be interested in ways to mask his own intent.

As described in Chapter IV, the ground truth chosen by the Red force is specified in the initialization files. The output of the COA perception files is completely independent of the ground truth and is calculated solely on observations of assets. From these asset counts, a posterior unit combination probability is calculated and compared to unit likelihoods on corridors. This provides the foundation for calculating the probability that the Red force is pursuing a particular course of action.

Each of the three courses of action as ground truth was replicated 30 times. It was necessary to produce multiple replications since the stochastic nature of the model produces fluctuating results. The mean and variance of the perception probabilities, as a function of time, for the ground truth course of action was extracted from the output files. These averages and a 95% confidence interval were then plotted against time to produce an average perception of the ground truth course of action. The conclusions drawn are based on one particular sensor. Changing the sensor variance could and should produce a different result.

1. COA 1

COA 1 was designed to provide an initial test of the model's ability to determine ground truth. All Red forces in COA 1 moved from the assembly areas (nodes 1 and 2) along the North Corridor towards the objective (node 19) as shown in Figure 9. Once the Red force has been detected, the Blue force reconnaissance is dispatched to the OP to begin sensor observations.

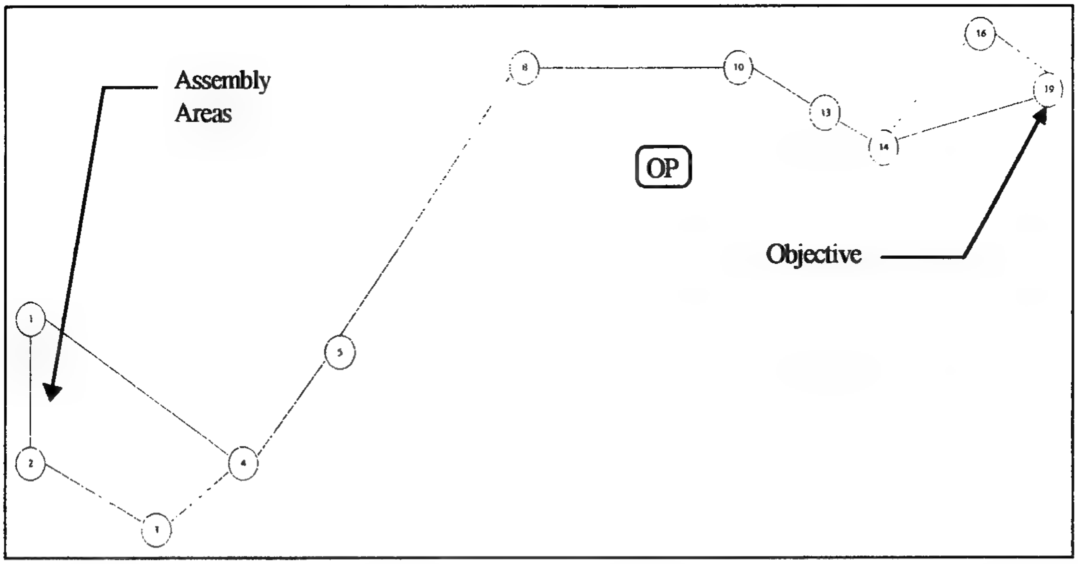

Figure 9. Red Force Battle Plan-COA 1

Each time a unit enters a physical node, a detection occurs. For nodes one through nine, this also triggers a mission for the reconnaissance helicopters. For the North Corridor, missions are initiated when units enter nodes one through five and node eight. The physical nodes, and transit nodes between physical nodes one and four, are common to all corridors and therefore, all courses of action. The transit node between nodes four and five is common to three of the four corridors. As a result, any single observation, by itself, taken before physical node five should not positively influence the perception towards ground truth.

For example, in the first replication, the first unit (the Red Tank Battalion) reaches node five at time 49.9 and proceeds towards node eight, where it is observed at time 51.1. COA 1 is the only course of action that has the Red Tank Battalion using this transit node. When the model updates the COA perception at time 52.0, the probability of COA 1 becomes 1.0. The fluctuation in reaching ground truth or probability of COA 1 equaling one is caused by two factors. First, when there is no observation mission in the queue and aircraft complete their orbit time, they return to the air base and do not assume another

mission until they have refueled. Second, units plan and depart physical nodes using normal distributions for generating times of occurrences of these events.

The first factor causes greater fluctuations than the second. Using the history files, it was found that sometimes units would cross physical nodes, thereby generating another observation mission before the helicopter completed orbit. Rather than depart, the helicopter would immediately begin the next sensor observation.

Given this fluctuation in perception probabilities, it was necessary to conduct more replications for each run and average the course of action perception probabilities. Figure 10 is a graph of the average perception probability using the best sensor (Alpha), over 30 replications, with a 95% confidence interval (shaded in gray). Vertical, or near vertical changes in the probability of COA 1 are indicative of critical observation periods (times 39, 51, 55, 72, and 76). The greater the change in probability, the more critical the observations. From the history file, these times can be traced to particular physical and transit nodes which are listed in Table 7. Also, the horizontal parts of the graph indicate parts of the network where reconnaissance assets are potentially wasted.

TABLE 7. CRITICAL NODES

Time	Node
39	Transit Node 6
51	Transit Node 9
55	Transit Node 9
72	Transit Node 9
76	Transit Node 9

Beyond the scope of this research, but certainly of interest, is the association between the network and the terrain. Determination of the critical nodes and times dictate the terrain areas where, and when, a commander would want reconnaissance.

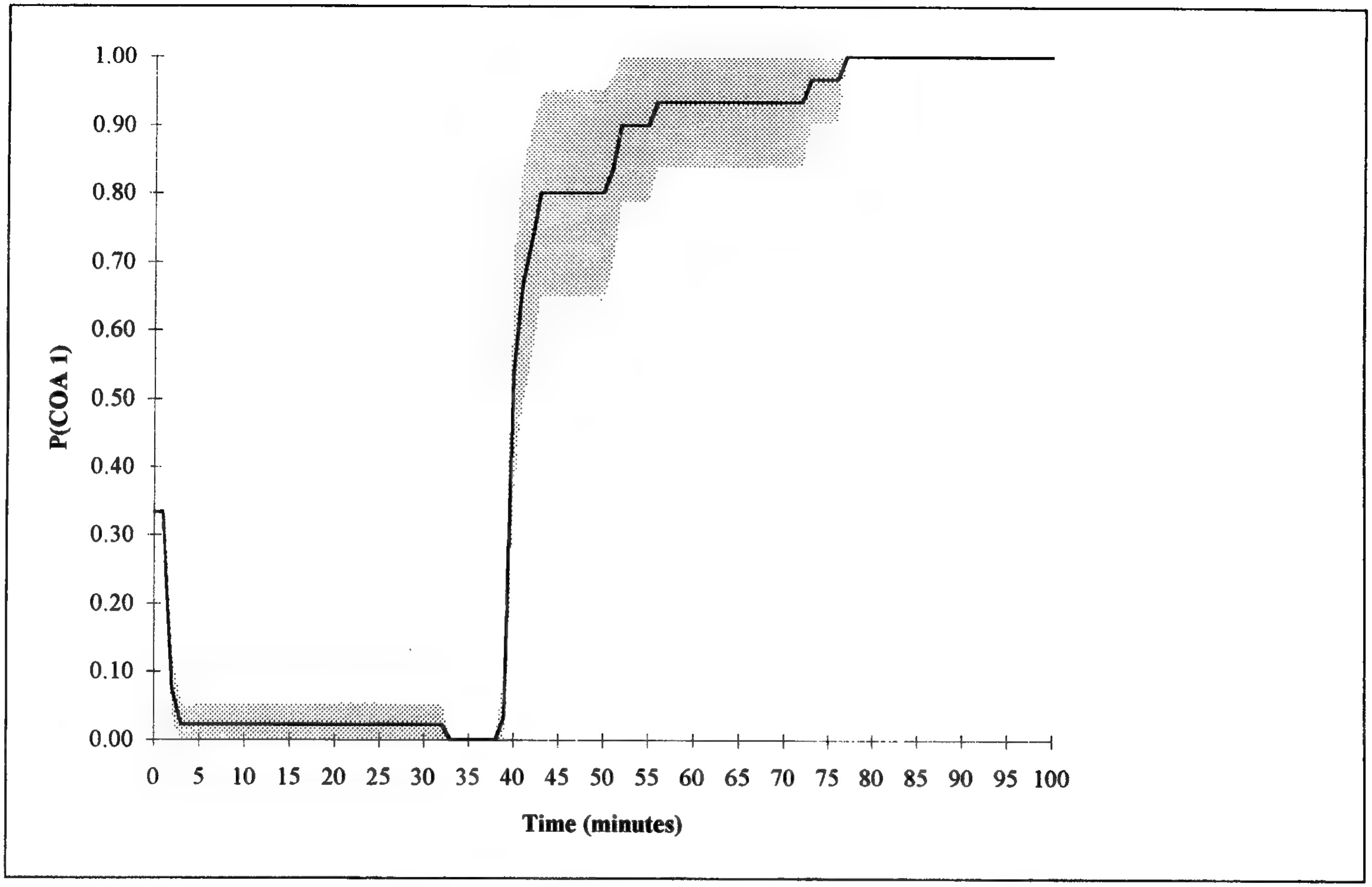

Figure 10. Average Perception of COA 1-Sensor Alpha

2. COA 2 and COA 3

Recall that these two COAs were designed to be nearly identical, with the main difference being the corridor taken by the Red Tank Battalion. Locating the tank battalion would determine ground truth. Figures 11 and 12 show the 30 replication averages for those runs. In each case, the COA probability plotted was the ground truth COA.

Unexpected in both cases was the cyclic pattern of the perception probabilities. When the two graphs are superimposed, the peaks and valleys are almost identical. There are two possible causes for this result. First, the weight of the prior probability is insufficient to overcome a poor observation. Second, the weight given to a current, exactly correct observation is insufficient to overcome the prior probability. In either case, this raises questions of how much prior information should be retained, how much weight to apply to that prior information, and whether the weights change, given the type of sensor being used.

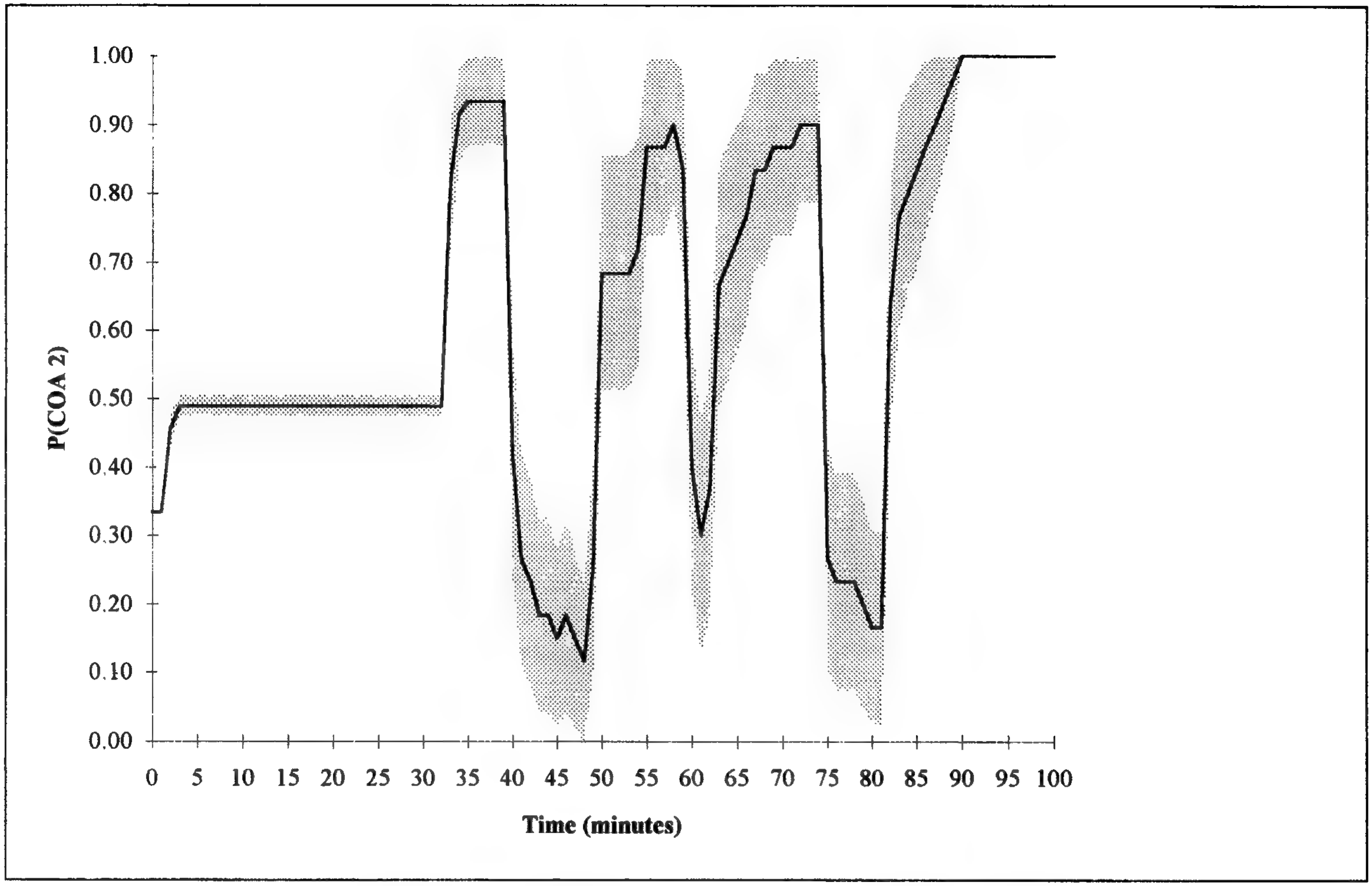

Figure 11. Average Perception of COA 2-Sensor Alpha

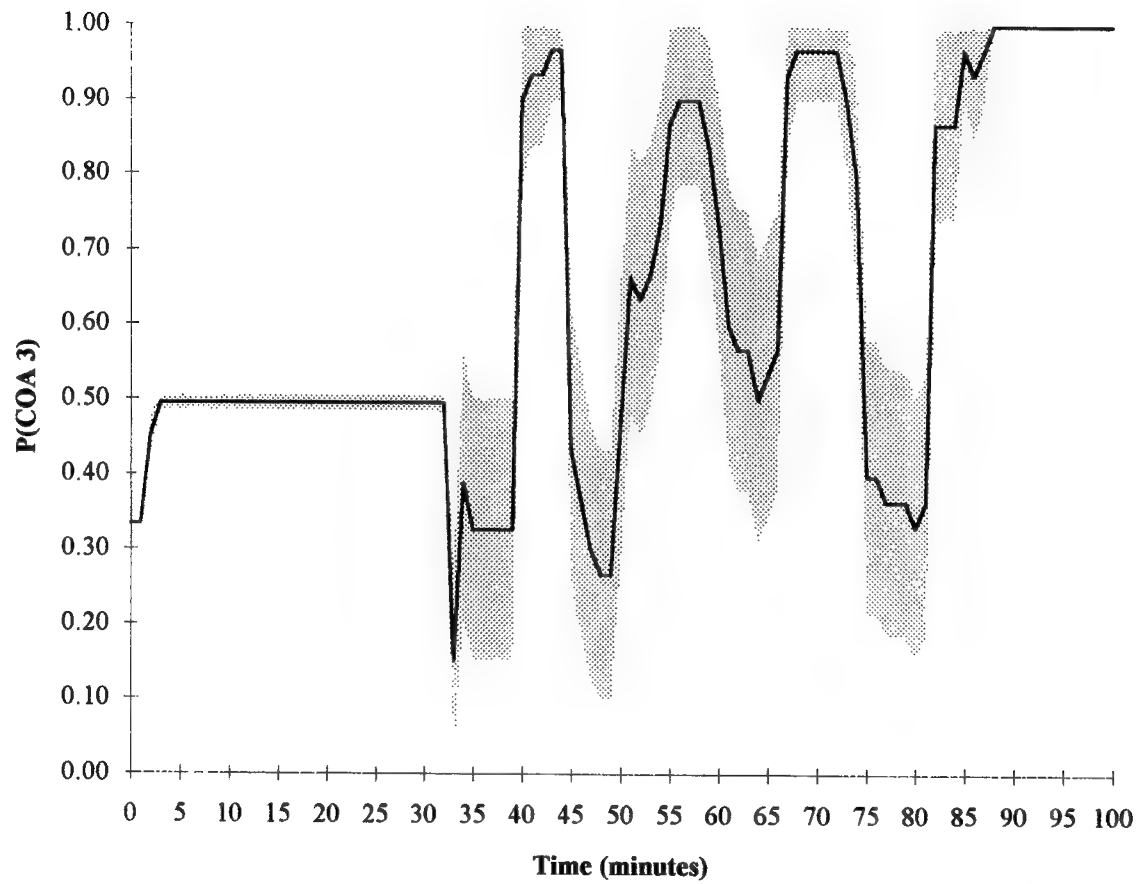

Figure 12. Average Perception of COA 3-Sensor Alpha

Assuming that the weights are correct, the history files were reviewed to determine possible alternative causes for the cyclic pattern. Specifically, causal determinations were made to find the observations that caused the perceptions to rise and fall so dramatically, and whether the changes in that perception produced the expected results. Recall from Table 6, that Sensor Alpha has a very compact performance distribution and will report near ground truth asset counts, which are then used to compute near perfect unit combinations. Therefore, the difference between the observed and expected unit combination is indistinguishable. Table 8 lists the Sensor Alpha observations that changed the COA perception for the first replication of COA 2. The table contains the update time, the ground truth unit combination (armor, artillery, mechanized), the observed node, the corridor(s) to which the observed node belongs (S=south, SC=south central), the computed COA probabilities, and the COAs that contain that unit combination on that corridor. Because of the sensor quality, there should be an expected shift towards the computed COA(s) that coincides with the "Likely COA."

TABLE 8. BREAKDOWN OF COA 2 REPLICATION

Update Time	Units	Observed Node	Corridors	Computed Probabilities			Likely COA
				COA 1	COA 2	COA 3	
33.00	(2,0,4)	Transit.05	SC,S	0.000	1.000	0.000	2,3
60.00	(1,3,2)	Transit.11	SC	0.000	0.000	1.000	2
60.00	(2,0,4)	Transit.12	S	0.000	0.000	1.000	2,3
66.00	(1,0,2)	Transit.13	SC	0.500	0.500	0.000	2
75.00	(1,3,2)	Transit.13	SC	0.000	0.000	1.000	2
82.00	(2,0,4)	Transit.16	S	0.000	1.000	0.000	2,3

For example, at update time 60.00, the sensor observation reports an asset count that coincides with a unit combination of one tank company, three artillery batteries, and two mechanized companies on transit node 11. From the history file, this unit combination corresponds to units Red.4 and Red.5. Transit node 11 belongs to the South Central Corridor. Course of action 2 is the only COA that has the unit combination listed above. The results show a definite problem linking the observation, and subsequent unit combination, with the correct COA(s). In this case, the probability shifted to COA 3. The last update observation was only sufficient to establish that COA 2 or 3 was the likely COA, and yet the model shifted to COA 2 with probability one.

To ensure that this was not a localized problem with Sensor Alpha, the runs were continued using the other two sensors with COA 2 as ground truth. Again, Sensors Bravo and Charlie, Figures 13 and 14, respectively, displayed the same fluctuating probabilities. In each of these two cases, as well as all other cases, there appeared to be no definite pattern; making it difficult to explain why the COA probabilities shifted the way they did. This method of comparison was used across several replications, with different combinations of sensors and ground truth COAs, each showing the same irregular results. These inconsistencies are documented here so that appropriate modifications can be made for the mature version of STLM. For this thesis, the reader should focus on the form of the results, not the specific numerical values.

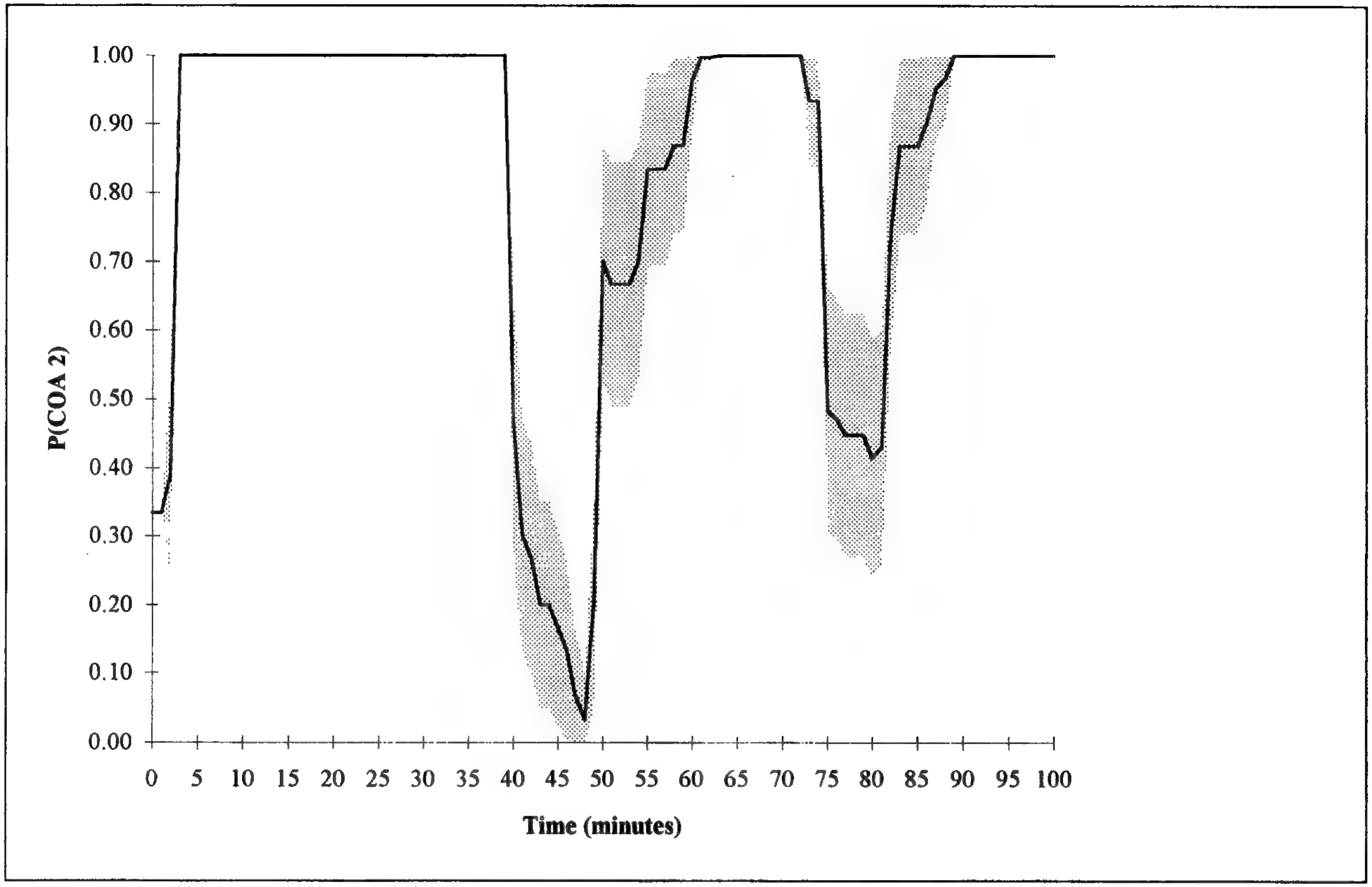

Figure 13. Average Perception of COA 2-Sensor Bravo

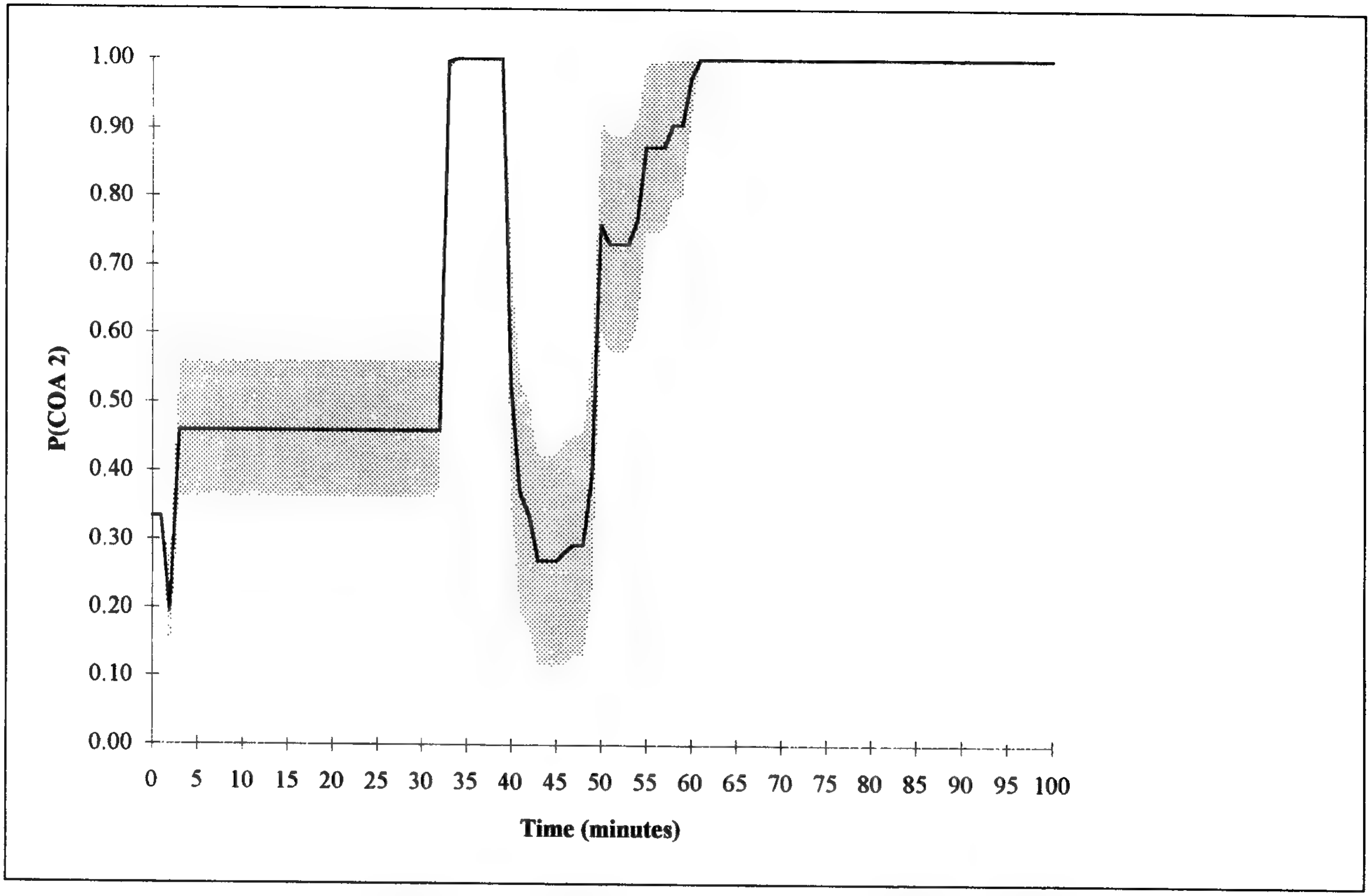

Figure 14. Average Perception of COA 2-Sensor Charlie

C. ATTRITION

There are many accepted attrition algorithms available. As described in Chapter III, the Bonder attrition equation was implemented in STLM. The unique aspect of the Bonder equation is that lethality is treated as a function of range. As assets get closer together, the lethality increases. Additionally, the scaling parameter, μ, (Equation (9)) can be manipulated to alter the impact that range has on lethality. For indirect fires, the parameter is set to zero, thereby eliminating the range effect on the attrition rate. The larger the value, the greater the impact that range has on the lethality of assets. In the following figures, side attrition is graphed as a function of time for each type asset.

In addition to the Bonder attrition methodology, there are several other parameters that impact on attrition. These are defined in Appendix A, STLM Initialization Data, and include:

- Minimum and maximum range of each asset.
- Priority allocation of fires to each asset.
- Direct fire versus indirect fire attrition rate.
- Unit orientation and unit array.
- Posture of units, defensive position versus open terrain.
- Cover and concealment afforded by the terrain.
- Speed of units on terrain.
- Rate of fire.
- Location of the Blue reserve unit.

Each of these can have a significant impact on the battle outcome. Effects of the unit orientation and the Blue reserve unit are evident in the graphs that follow. Both of these

factors influence the graphs as a change in the rate of attrition of specific assets. The attrition rates, scaling parameter, and allocation priority for each asset were taken from a Land Combat class project. [Ref 9:p. 5-7] Recall that the Blue reserve company is committed to supporting the defense of the corridor associated with Blue's perception of Red's most likely COA. In all cases, Blue correctly perceived Red's ground truth COA and the reserve force was committed to defending the appropriate corridor.

Because the attrition model is deterministic, the results for each COA were the same, regardless of the sensor used. There are many parameters outside of the Bonder range dependency equation that impact on the attrition of assets. Given that the model does not currently provide item level resolution, combined with the unit resolution output, the specific details and attrition patterns are not easily defined.

For example, the attrition of artillery assets, with scaling parameter set to zero, was extremely sensitive to any change in pK, rate of fire, and priority allocation. In the graphs, Blue artillery was always attrited to breakpoint asset strength unless the Red artillery pK was set arbitrarily low. Conversely, the exact opposite was true of the Red artillery. The purpose of this research was not to determine pK values that made the graphs appear understandable; as such, the original values were retained and this is left as an area requiring further study.

1. Attrition for COA 1

Figures 15 and 16 show the attrition of Blue and Red assets, respectively, as a function of time for COA 1. Since COA 1 has Red units only on the North Corridor (Figure 9), there is only one arc-node path into the defensive perimeter. As a result, Blue and Red attrite each other as Red comes into the direct fire range of one Blue unit at a time. Other observations on Blue's attrition are listed below.

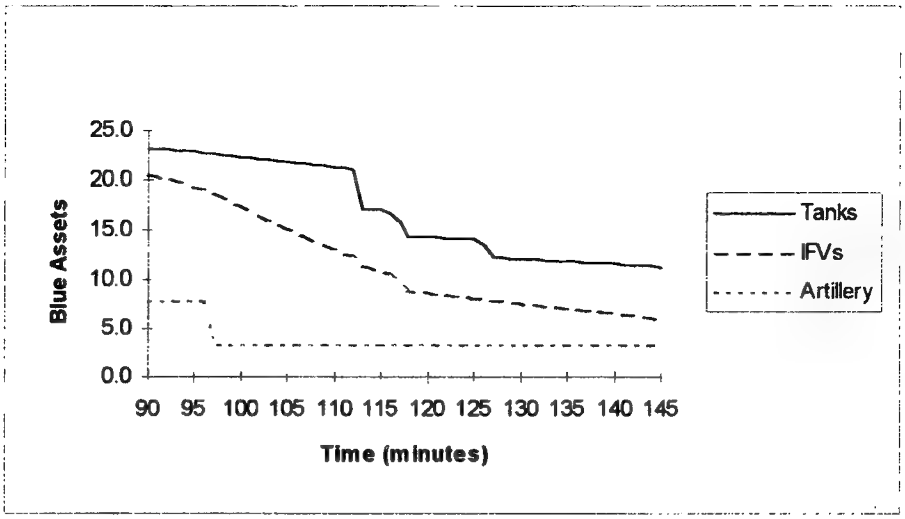

Figure 15. Blue Attrition for COA 1

- The first significant change in Blue tank strength is the result of the Red Tank Battalion coming into direct fire range at times 110 through 115 (Figure 15).
- The attrition of the Red Tank Battalion at times 112 through 117 is the result of the Blue reserve unit being committed (Figure 16).
- The lack of significant BMP attrition is the possible result of inadequate allocation of Blue firers (Figure 16).

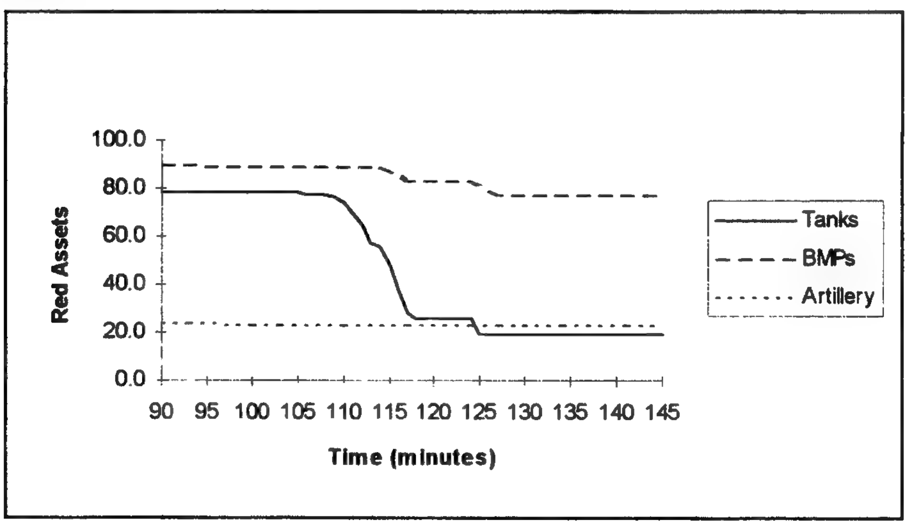

Figure 16. Red Attrition for COA 1

2. Attrition for COA 2

For COA 2, the new attrition patterns, shown in Figures 17 and 18, are primarily the result of the different orientation of Red and Blue forces. Recall, that in this course of action, three of the four corridors are being used by the Red force. This leads to an almost one-on-one orientation of units.

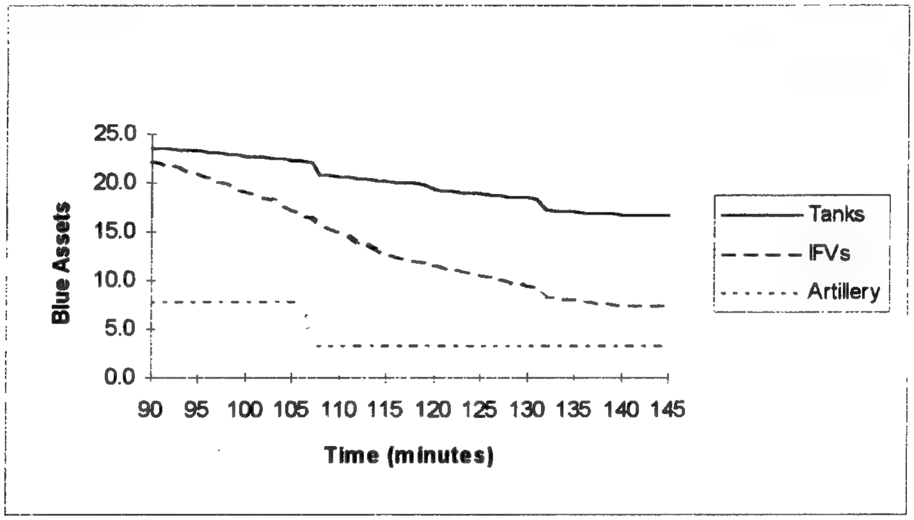

Figure 17. Blue Attrition for COA 2

- The percentage of Blue tanks surviving is much greater, given this course of action versus COA 1 (Figure 17).
- The Blue reserve force is committed at time 130 resulting in the final major loss of Red tank assets (Figure 18).
- By taking the North Central Corridor, the Red Tank Battalion comes into direct fire range with the center Blue company first, at time 105; and then the two flank Blue companies, at times 110 through 120 (Figure 18).

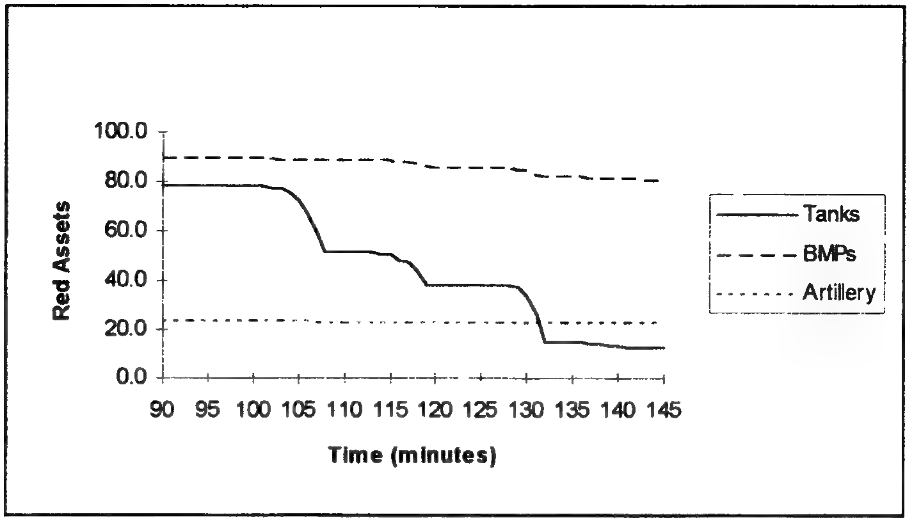

Figure 18. Red Attrition for COA 2

3. Attrition for COA 3

Figures 19 and 20 show Blue and Red attrition, respectively, for COA 3.

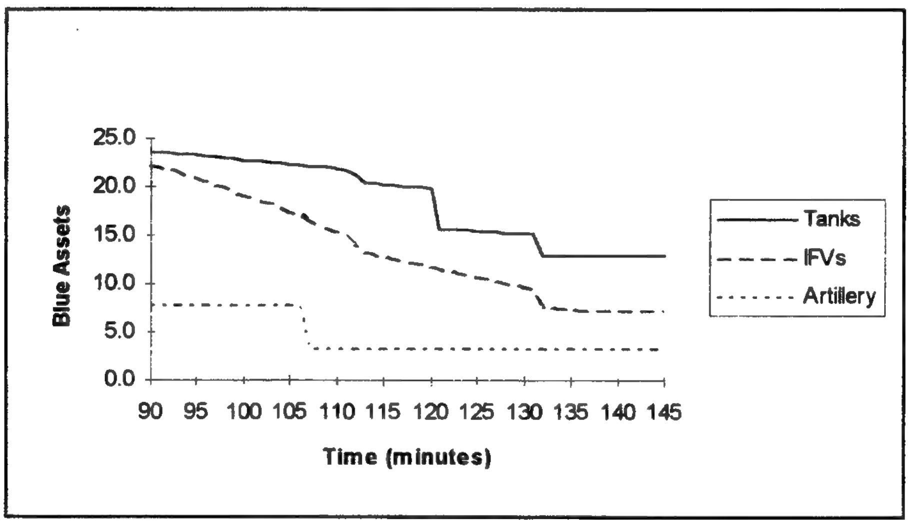

Figure 19. Blue Attrition for COA 3

- The attrition of Blue tanks occurs in two specific stages. At time 120, Red units from the South Central Corridor come in direct fire range and at time 130 Red units from the South Corridor enter the battle (Figure 19).
- As before, the Blue reserve unit commits at time 128, leading to the final attrition of Red tanks and BMPs (Figure 20).

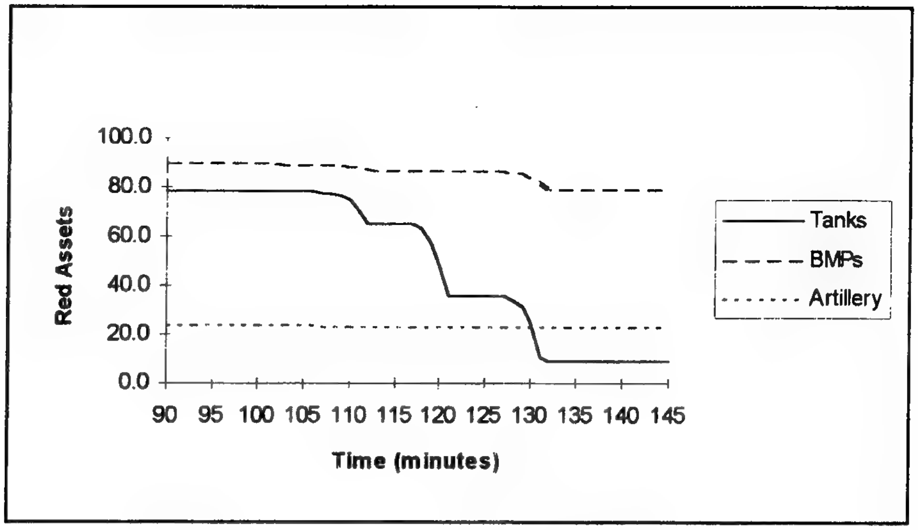

Figure 20. Red Attrition for COA 3

4. Comparison of End Strength Percentages

Table 9 lists the end strength percentages for each side as a function of beginning strength assets. Assuming that all of the other parameters are valid, the results would indicate that the Blue force should consider repositioning units to improve the percentage of assets surviving based on which COA is perceived to be the ground truth.

TABLE 9. PERCENT OF ASSETS REMAINING

		COA 1	COA 2	COA 3
	Tanks	45	71	54
BLUE	IFVs	25	33	29
	Artillery	38	38	38
	Tanks	24	17	13
RED	BMPs	86	89	88
	Artillery	96	96	96

After repositioning units in the model, the simulation could be run again to determine new outcomes.

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

STLM has the potential to be the first stochastic perception model that can be put in the hands of the small theater commander without requiring a support team for operation. This research was a preliminary analysis into that concept. Compared to other high resolution models such as CASTFOREM and JANUS(A), STLM can assist the analyst in evaluating different force structures, sensor platforms, and other uncertainties associated with combat. The COA perception methodology is an invaluable tool for any military decision maker.

The method of decreasing the detection rate in the initialization files proved to be the easier, if not correct choice in maintaining detections without influencing the COA perception update. By leaving the detection routine in the model, continuous type reconnaissance can be coupled with periodic intelligence gathering assets. Using only the sensor observations yielded the expected outcomes: the smaller the sensor standard deviation, the more accurate the model was at predicting the correct unit combination. From this standpoint alone, STLM provides the analyst with an invaluable tool to compare the performance of different sensors.

In all cases, STLM was able to correctly determine the Red force's ground truth course of action. The model did produce greater than expected fluctuations in probabilities during COA updates, indicating that the prior probability was not sufficiently weighted or that observations were being bundled prior to a perception update. The model did not meet all expectations in the COA perception update, and needs more development before it becomes an acceptable tool for the analyst.

Both sub-elements of the of the maneuver model, ground and air, performed as would be expected. Red force ground units moved in accordance with specified corridor and course of action assignments. The Blue force reserve was dispatched to the correct position and location. The reconnaissance helicopters not only provided the required asset counts for the COA updates, but continued to observe when new missions entered the queue.

The attrition of units was consistent with the Bonder range dependency equation. The rate of attrition increased as units came closer together. As is common in most low resolution models, STLM currently lacks the collective detail to draw any strong conclusions about force positioning, rates of fire, and individual performance of units. Determining which unit caused an opposing unit to be attrited is not easily obtained from the output files. At best, determining when units come into direct fire range can only be estimated by creating the attrition graphs and comparing them to the ground unit history file. Currently, there are no sector of fire rules in the attrition model. Therefore, units engage all units within range and not just those in an assigned sector.

B. RECOMMENDATIONS

The following recommendations are divided into two sections and should be the major focus for the continuing development of STLM. The first section describes required changes in the programming:

- Incorporate rule sets into the maneuver and attrition models giving units specific sectors of fire along arc-node boundaries.

- Amplify the COA updating in the output files to ensure that no observations are bundled prior to the COA update cycle.

- Incorporate a variable weight parameter, as part of the initialization files, to allow weighting of the prior COA probability.

- Direct all attrition results to the attrition output file.

The second set of recommendations focuses on areas that need further investigation:

- Evaluate the COA weighting parameter described above.

- Determine suitable bounds on the Bonder range parameter for various weapon systems.

- Verification of COA perception updating algorithm, with validation of information fusion.

- Expansion of helicopter representation.

- Accuracy of sensor representation.

- Accuracy of units represented (scenario representation).

- Effects of terrain and weather as they relate to the C3I, maneuver, and attrition models.

- Possibility of item level resolution.

After a making the recommended code changes, verifying the complete STLM code, and investigating the parameters listed above, the final focus should be directed at validating the STLM results with NTC results.

APPENDIX A. STLM DATA RECORDS

A. OVERVIEW

These instructions were written by Harold Yamauchi. [Ref. 10] The program obtains scenario data from an ASCII text file. This file can be identified by a .NET file extension. There are currently twenty-eight types of data that may be entered in the .NET file. These data must be entered in the following order:

1. Parameter
2. Side
3. Relationships
4. Factions
5. Physical Node
6. Arc
7. Observation Points
8. Equipment Types
9. Combat Systems
10. Combat System pK Table
11. Air Munitions
12. Ground Munitions
13. Munitions Sticks and Volleys
14. Ground-Air Weapons
15. Radar
16. Type I Sensor
17. Jammer
18. Altitude Bands
19. Aircraft
20. Air Defense/Fire Support Type
21. Air-Ground pK Table
22. Ground-Ground pK Table
23. Atom Type
24. Combat Unit
25. Air Base
26. Squadron
27. Corridor
28. Course of Action

As the model evolves, new data will be identified and added to the scenario data file and old data that are no longer required will be dropped.

The records are described below. Each record must begin on a new line. Aside from these restrictions, the record fields are freely formatted. Each record itself is allowed to occupy one or more lines as shown by the sample data file in paragraph B. An asterisk (*) is used to mark the end of one type of data and the beginning of the next type. Comments may be entered to the right of the * delimiter as shown in the sample data file. Note that the comments are restricted to the line containing the * delimiter.

B. INSTRUCTIONS FOR INITIALIZATION FILE

1. Parameters

Enter the following data to establish the network limits and environment:

a. Coordinate system
   1 = latitude/longitude
   2 = Cartesian
b. Random number generator seed - integer >= 0.
c. Time step - real in minutes.
d. Weather over battlefield
   0 = Clear
   1 = Rain/snow
   2 = Heavy clouds, no precipitation
e. Surprise interval - real in minutes. Amount of time a unit is allowed to remain surprised.
f. Sensor cycle time - real in minutes. Time allowed for sensor observations to be made before updating perceptions.
g. Battlefield/air network boundaries. If the coordinate system is latitude/longitude, enter the following:
   (1) Upper left latitude of air network overlay - 2 to 9 characters. Entered in degrees-minutes-seconds format with the last character indicating the direction, "N" = north and "S" = south, from the equator. For example, 30-20-10N. The degrees, minutes, and seconds must be separated by the dash "-". Do not embed blanks. If seconds is zero, the seconds field may be omitted, e.g., 30-20N. If seconds and minutes are zero, the minutes and seconds fields may be omitted, e.g., 30N. Latitudes may range from 90S to 90N
   (2) Upper left longitude of air network overlay - 2 to 10 characters. Entered in degrees-minutes-seconds format with the last character indicating the direction, "E" = east and "W" = west, from the prime meridian. For example, 140-30-20E. The degrees, minutes, and seconds must be separated by the dash "-". Do not embed blanks. If seconds is zero, the seconds field may be omitted, e.g., 140-30E. If seconds and minutes are zero, the minutes and seconds fields may be omitted, e.g., 140E. Longitudes may range from 180W to 180E.
   (3) Lower right latitude of air network overlay - 2 to 9 characters. Format is similar to the upper left latitude of air network overlay described in 6.a.(1), above.
   (4) Lower right longitude of air network overlay - 2 to 10 characters. Format is similar to the upper left longitude of air network overlay described in 6.a.(2), above.

If the coordinate system is 2 (Cartesian), the minimum x- and y-coordinates are assumed to be zero (0) and not entered. Enter the following:

(1) Maximum x-coordinate of air network overlay - real in kilometers.
(2) Maximum y-coordinate of air network overlay - real in kilometers.

h. Air grid length - real > 0. The air network consists of square grids. The number entered here is the length of a side of a grid in kilometers. The program uses this number and the preceding coordinates in 6 to determine the number of rows and columns of grids in the air network.
i Step size for air defense lethal area calculations - real > 0. An air defense unit's assets may cover all or part of an air grid. The area of the grid that is covered by these assets is calculated numerically. The step size determines the accuracy of the calculation. As the step size decreases, the accuracy of the calculation increases, but at a cost of increased calculation time. For grids that are at least 10 Km in length a step size of 0.01 should provide sufficient accuracy.
j. Air grid lethality multiplier - real >= 0.
k. Air distance multiplier - real >= 0.
l. Round effects method
   1 = Independent shots, no effects overlap
   2 = Confetti 1 approximation, effects overlap

Parameter data delimiter - "*"

2. Side Records

For each side enter the following:

a. Side name - 1 to 10 characters. Do not embed blanks.
b. Side color
   0 = White
   1 = Blue
   2 = Red
c. Reserved - integer. Enter a one (1).
d. Reserved - integer. Enter a one (1).
e. Reserved - integer. Enter a one (1).
f. Reserved - real. Enter a one (1.0).
g. Reserved - real. Enter a one (1.0).
h. Reserved - real. Enter a one (1.0).
i. Reserved - real. Enter a one (1.0).
j. Reserved - real. Enter a one (1.0).
k. Reserved - real. Enter a one (1.0).
l. Reserved - real. Enter a one (1.0).
m. Reserved - real. Enter a one (1.0).
n. Reserved - real. Enter a one (1.0).

o. Reserved - real. Enter a one (1.0).
p. Reserved - real. Enter a one (1.0).
q. Reserved - real. Enter a one (1.0).
r. Reserved - real. Enter a one (1.0).
s. Reserved - real. Enter a one (1.0).
t. Reserved - real. Enter a one (1.0).
u. Reserved - real. Enter a one (1.0).
v Plan wait period - real in minutes. Interval between the time a unit arrives at a physical node and the time that unit starts planning.
w Mean time to depart physical node - real in minutes. Used to determine the time a unit should depart a physical node. A scheduled departure time is determined by drawing from a normal distribution whose mean is equal to this data element and whose standard deviation is equal to one-tenth of the mean.
x. Departure deviation time - real in minutes. Used to determine the time a unit will actually depart a physical node. The amount of deviation is obtained by drawing from a normal distribution whose mean is equal to this data element and whose standard deviation is equal to one-tenth (0.1) of the mean. The deviation is added to the scheduled departure time.
y. Reserved - real. Enter a one (1.0).
z. Reserved - real. Enter a one (1.0).
aa. Reserved - real. Enter a one (1.0).
bb. Air defense coverage update period - real > 0. Time (minutes) between updates of enemy air defense coverage.
cc. Scheduled mission taxi time - real > 0 in minutes.
dd. Reactive mission taxi time - real > 0 in minutes.
ee. Recovering mission taxi time - real > 0 in minutes.
ff. Abort mission if a flight is partially destroyed on the ground.
    0 = No
    1 = Yes
gg. Flights search for secondary configuration if there is insufficient ammo to launch the mission.
    0 = No
    1 = Yes
hh. Flight group join-up time - real > 0 in minutes.
ii. Reconnaissance mission on-station time - real > 0 in minutes.
jj. Waiting time before a delayed mission is canceled - real > 0 in minutes.

Side data delimiter - "*"

3. Relationships

The data entries described here are a first attempt to provide simple rules that each side applies during planning. It is also to be used when an encounter occurs between units from different sides. Each record consists of three fields described below.

a. Side name - 1 to 10 characters. The name of a previously defined side.
b. Side name - 1 to 10 characters. The name of another previously defined side.
c. Relationship. This code determines the action taken by each side. No action is taken against a neutral (0) or friendly (1) unit. Combat only occurs between the belligerent sides (2).
   0 = neutral
   1 = friend
   2 = foe

Relationship data delimiter - "*"

4. Faction Records

For each faction, enter the following:

a. Faction name - 1 to 10 characters. Do not embed blanks.
b. Initial side - 1 to 10 characters. The name of a previously defined side. This is the side this faction aligns with at time 0.
c. Atom size - character. Enter one of the following:
   PLATOON
   COMPANY
d. Atom's parent size - character. Enter one of the following:
   COMPANY (if atom size is PLATOON)
   BATTALION (if atom size is COMPANY)

Faction data delimiter - "*"

5. Physical Node Records

For each physical node, enter the following:

a. Physical node name - 1 to 10 characters. Do not embed blanks.
b. Node ID number - integer.
c. Location. If the coordinate system (see Parameters, item 1) is 1 (latitude/longitude), enter the following:
   (1) Latitude - 2 to 9 characters. Entered in degrees-minutes-seconds format with the last character indicating the direction, "N" = north and "S" = south, from the equator. For example, 30-20-10N. The degrees, minutes, and seconds must be separated by the dash "-". Do not embed blanks. If

seconds is zero, the seconds field may be omitted, e.g., 30-20N. If seconds and minutes are zero, the minutes and seconds fields may be omitted, e.g., 30N. Latitudes may range from 90S to 90N.

(2) Longitude - 2 to 10 characters. Entered in degrees-minutes-seconds format with the last character indicating the direction, "E" = east and "W" = west, from the prime meridian. For example, 140-30-20E. The degrees, minutes, and seconds must be separated by the dash "-". Do not embed blanks. If seconds is zero, the seconds field may be omitted, e.g., 140-30E. If seconds and minutes are zero, the minutes and seconds fields may be omitted, e.g., 140E. Longitudes may range from 180W to 180E.

If the coordinate system is 2 (Cartesian), enter the following:

(1) X-coordinate - real in kilometers.
(2) Y-coordinate - real in kilometers.

d. Diameter - real in kilometers.

e. Terrain use
   1 = air base
   2 = logistics base
   3 = defensive point
   4 = obstacle
   5 = arc crossing point

f. Terrain
   0 = sea
   1 = open - no defenses
   2 = hasty defenses
   3 = deliberate defenses
   4 = major obstacle (used when node represents a major obstacle)
   5 = urban

g. Capacity - real. Number of action units that can simultaneously occupy the node.

h. Obstacles
   0 = none
   1 = minefield
   2 = not defined
   3 = not defined
   4 = chemical contamination
   5 = radiological contamination

i. Cover - real. Amount of cover/concealment at node. Entered as a fraction [0.0, 1.0].

j. Suitable for concealed approach.
   0 = No
   1 = Yes

k. Suitable for defensive obstacles.
   0 = No

1 = Yes

l. Reserved - integer. Enter a zero (0).
m. Reserved - real. Enter a zero (0.0).
n. Reserved - real. Enter a zero (0.0).
o. Reserved - real. Enter a zero (0.0).
p. Reserved - real. Enter a zero (0.0).
q. Reserved - integer. Enter a zero (0).

Node data delimiter - "*"

6. Arc Records

For each arc, enter the following:

a. Source node - 1 to 10 characters. The name of a physical node.
b. Destination node - 1 to 10 characters. The name of a physical node.
c. Number of transit nodes - integer >= 1
d. Transit node information. For each transit node, enter the following:
   (1) Transit node name - 1 to 10 characters. Do not embed blanks.
   (2) Distance - real in kilometers.
   (3) Road type.
      1 = primary
      2 = secondary
      3 = unpaved/trail
   (4) Terrain.
      0 = sea
      1 = flat
      2 = rolling
      3 = severe
   (5) Wetland/marsh
      0 = No
      1 = Yes
   (6) Natural obstacle
      0 = No
      1 = Yes
   (7) Manmade obstacle
      0 = No
      1 = Yes
   (8) Mountain
      0 = No
      1 = Yes
   (9) Urban
      0 = No
      1 = Yes
   (10) Trafficability

1 = No restriction
2 = Road movement only
3 = No heavy equipment
4 = No wheeled vehicles
5 = Foot only

(11) Capacity - real. Width in kilometers across mobility corridor.

(12) Obstacles
0 = none
1 = minefield
2 = requires bridging
3 = requires physical clearing (non-explosive)
4 = chemical contamination
5 = radiological contamination

(13) Cover - real. Amount of cover/concealment at node. Entered as a fraction in the range [0.0, 1.0]

(14) Suitable for ambush
0 = No
1 = Yes

(15) Suitable for obstacles
0 = No
1 = Yes

(16) Reserved - integer. Enter a zero (0).

(17) Detection rates. For each side enter the following:
(a) Side name - 1 to 10 characters. The name of a previously defined side.
(b) Detect rate - real >= 0. Rate (per hour) at which this side detects units.

Arc data delimiter - "*"

7. Observation Point Records

For each observation point, enter the following:

a. Side name - 1 to 10 characters. The name of a previously defined side.
b. Observation point name - 1 to 10 characters. Do not embed blanks.
c. Primary node - 1 to 10 characters. The name of a previously defined physical node. Detections at this node will trigger reconnaissance missions to be sent to this observation point.
d. Maximum number of detections allowed at primary node - integer.
e. Location. If the coordinate system (see Parameters, item 1) is 1 (latitude/longitude), enter the following:
   (1) Latitude - 2 to 9 characters. Entered in degrees-minutes-seconds format with the last character indicating the direction, "N" = north and "S" = south, from the equator. For example, 30-20-10N. The degrees, minutes, and seconds must be separated by the dash "-". Do not embed blanks. If seconds is zero, the seconds field may be omitted, e.g., 30-20N. If seconds

and minutes are zero, the minutes and seconds fields may be omitted, e.g., 30N. Latitudes may range from 90S to 90N.

(2) Longitude - 2 to 10 characters. Entered in degrees-minutes-seconds format with the last character indicating the direction, "E" = east and "W" = west, from the prime meridian. For example, 140-30-20E. The degrees, minutes, and seconds must be separated by the dash "-". Do not embed blanks. If seconds is zero, the seconds field may be omitted, e.g., 140-30E. If seconds and minutes are zero, the minutes and seconds fields may be omitted, e.g., 140E. Longitudes may range from 180W to 180E.

If the coordinate system is 2 (Cartesian), enter the following:

(1) X-coordinate - real in kilometers.

(2) Y-coordinate - real in kilometers.

f. Number of additional physical and transit nodes to observe - integer >= 0.

g. Node list. For each physical and transit node that will be observed from this observation point, enter the Node name - 1 to 10 characters. The name of a previously defined physical or transit node.

Observation Point data delimiter - "*"

8. Equipment Type Records

For each equipment type, enter the following:

a. Equipment name - 1 to 10 characters. Do not embed blanks.

b. Classification - integer. Enter a one (1).

c. Strength scores. FTLM utilizes twelve strength index numbers to measure the "potential" of a combat unit. Each equipment is assigned a score and each score is weighted by the number of assets in the unit. Each index number is obtained by summing the appropriate weighted scores. (The Asset/Strength Flag Array described in e. determines which scores can be used to obtain the index number.)

d Soft/hard target flag

1 = soft target

2 = hard target

e. Asset/strength flag array. The flags are entered in a 1 x 12 array. Each flag can be set to one of two values. If the flag is set to zero (0) in cell I of the array, the equipment's score will not be used to calculate strength index I. If the flag is set to one (1) in cell I, the equipment's score will be used to calculate strength index I. The cells of the array are from left to right:

(1) Ground to ground attrition strength

(2) Ground to air attrition strength

(3) Air to ground attrition strength

(4) Air to air attrition strength

(5) C^2 C^3I strength

(6) Communication C^3I strength

(7) Intelligence C^3I strength
(8) Counter-C^3I measures C^3I strength
(9) Ground support logistics strength
(10) Air support logistics strength
(11) Ammunition logistics strength
(12) POL logistics strength

Equipment type data delimiter - "*"

9. Combat System Records

For each combat system enter the following:

a. Number of combat systems - integer.
b. Combat system information. For each combat system, enter the following:
   1) Combat system name - 1 to 10 characters. Do not embed blanks.
   (2) Combat system's equipment type name - 1 to 10 characters. The name of a previously defined equipment type.
   (3) Weapon classification
       1 = direct fire
       2 = area fire
   (4) Minimum weapon engagement range - real in meters.
   (5) Maximum weapon engagement range - real in meters.
   (6) Bonder range parameter - real
   (7) Rate of fire - real. Rounds per minute.

Combat System data delimiter - "*"

10. Combat System pK Records

For each combat system, enter the following:

a. Name of firing combat system - 1 to 10 characters. The name of a previously defined combat system.
b. Number of target combat systems - integer.
c. pK information. Enter the following for each target combat system:
   (1) Name of target combat system - 1 to 10 characters. The name of a previously defined combat system.
   (2) Firing priority - real in the range [0.0, 1.0]. This is the fraction of allocated rounds that are fired at the target.
   (3) Probability of kill/lethal area - real. If the system is a direct fire weapon, this is a single shot probability of kill. If the system is an area fire weapon, this is the lethal area of the weapon.

Combat System pK data delimiter - "*"

11. Air Munitions Records

For each munitions, enter the following:

a. Munitions name - 1 to 10 characters. Do not embed blanks.

b Function

1 = air-air

2 = air-ground

3 = anti-radiation missile

4 = self-protect weapon

5 = mine

Air munitions data delimiter - "*"

12. Ground Munitions Records

For each munitions, enter the following:

a. Munitions name - 1 to 10 characters. Do not embed blanks.

b. Function. Ground munitions will be delivered to their targets through FTLM's air network. A ballistic weapon takes the most direct route to its target while a terrain-following weapon may follow a path that minimizes the effects from enemy air defenses.

1 = not defined

2 = not defined

3 = not defined

4 = not defined

5 = not defined

6 = not defined

7 = ballistic

8 = terrain following

c. Reserved - integer. Enter a one (1).

d. Round speed - real in meters per minute.

e. Maximum range - real in meters.

Ground munitions data delimiter - "*"

13. Munitions Stick and Volley Records

For assessment purposes, air munitions are grouped into sticks; ground munitions are grouped into analogous volleys. For each munitions stick or volley, enter the following:

a. Stick or volley name - 1 to 10 characters. Do not embed blanks.

b. Munitions name - 1 to 10 characters. The name of a previously defined air or ground munitions.

c. Number of munitions - integer. Quantity of this munitions in this stick or volley.
d. Soft target radius of effect - real in meters.
e. Hard target radius of effect - real in meters.
f. Stand-off range - real in meters.

Munitions stick and volley data delimiter - "*"

14. Ground-Air Weapon Records

For each ground-air weapon, enter the following.
a. Weapon name - 1 to 10 characters. Do not embed blanks.
b. Round speed - real in meters per minute.
c. Minimum range - real in meters.
d. Maximum range - real in meters.
e. Maximum altitude - real in meters.

Ground-air weapon data delimiter - "*"

15. Radar Records

For each radar, enter the following:
a. Radar name - 1 to 10 characters. Do not embed blanks.
b. Radar range - real. Range in meters against a one square meter target.
c. Radar altitude - real. Altitude in meters against a one square meter target.
d. Fire control capability - integer. Number of fire control radar that may be controlled by this radar acting as an acquisition radar.
e. Number of TELs per fire control radar - integer. Number of transporter/erector/launchers (TELs) this radar can control as a fire control radar.
f. Capable of unqueried acquisition
   0 = No
   1 = Yes
g. Sector sweep angle - real in degrees [0, 360].
h. Minimum elevation angle - real in degrees [0, 45].
i. Fire control range - real in meters.
j. Maximum operating range - real in meters.
k. Netted acquisition time - real in seconds.
l. Unnetted acquisition time - real in seconds.

Radar data delimiter - "*"

16. Type I Sensor Records

Type I sensors report the number of assets observed at a physical or transit node.

For each sensor, enter the following:

a. Side name - 1 to 10 characters. The name of a previously defined side.
b. Sensor name - 1 to 10 characters. Do not embed blanks.
c. Number of equipment types the sensor can see - integer.
d. Equipment list. For each equipment type the sensor can see, enter the equipment name - 1 to 10 characters. The name of a previously defined equipment type.
e. Number of physical nodes - integer. This number should equal the number of physical nodes defined in the Physical Node Records.
f. Physical node information. For each physical node, enter the following:
   (1) Physical node name - 1 to 10 characters. The name of a previously defined physical node.
   (2) Sensor standard deviation by equipment type - real > 0. Enter the sensor's standard deviation for each equipment type listed in h. These standard deviations must be listed in the same order as the equipment types appear in h.
g. Number of transit nodes - integer. This number should equal the number of transit nodes defined in the Arc Records.
h. Transit node information. For each transit node, enter the following:
   (1) Transit node name - 1 to 10 characters. The name of a previously defined transit node.
   (2) Sensor standard deviation by equipment type - real > 0. Enter the sensor's standard deviation for each equipment type listed in h. These standard deviations must be listed in the same order as the equipment types appear in h.

Type I sensor data delimiter - "*"

17. Jammer Records

For each jammer, enter the following:

a. Jammer name - 1 to 10 characters. Do not embed blanks.
b. Number of radar - integer. Number of radar that are vulnerable to the jammer.
c. Jammer effect information. For each radar that can be degraded by this jammer, enter the following:
   (1) Radar name - 1 to 10 characters. The name of a previously defined radar.
   (2) Jammer effect - real. Decrease in radar's 1 m^2 burn through range with jammer turned on (meters). If the jammer effect is zero, the radar and its jammer effect do not have to be listed. The FTLM program assumes that the jammer is ineffective against any unlisted radar.

Jammer data delimiter - "*"

18. Altitude Records

This is a departure from TAC Thunder In TAC Thunder, altitude bands are an aircraft characteristic. Since the algorithm that assesses ground-to-air outcomes requires the mission package (flight group) altitude, if a package is composed of more than one type of aircraft and the altitude bands are different for each type, how is the package's altitude determined? Assume for now that for a given side, the altitude bands are the same for all aircraft belonging to that side. For each side, enter the following:

a. Side name - 1 to 10 characters. The name of a previously defined side.
b. Low dash altitude - real in meters.
c. Low penetration altitude - real in meters.
d. High dash altitude - real in meters.
e. High penetration altitude - real in meters.
f. High cruise altitude - real in meters.
g. Orbit altitude - real in meters.

Altitude data delimiter - "*"

19. Aircraft Records

For each aircraft, enter the following:

a. Aircraft name - 1 to 10 characters. Do not embed blanks.
b. Fixed-wing flag
   0 = Rotary-wing
   1 = Fixed-wing
c. Naval capable
   0 = No
   1 = Yes
d. Squadron size - integer. Number of aircraft of this type flown by a squadron.
e. Radar name - 1 to 10 characters. The name of the radar carried by this aircraft and defined in the Radar Records. If no radar is carried, enter "NONE".
f. Low dash altitude speed - real in knots.
g. Low penetration altitude speed - real in knots.
h. High dash altitude speed - real in knots.
i. High penetration altitude speed - real in knots.
j. High cruise altitude speed - real in knots.
k Orbit altitude speed - real in knots If the aircraft does not fly orbiting missions, enter a zero (0).

l. Radar cross section - real in square meters and measured 20° off the nose.
m. Takeoff runway length - real in feet
n. Landing runway length - real in feet.
o. Minimum flight size - integer
p. Probability of short term repair - real [0, 1]. The sum of this probability and the probability of long term repair must not exceed one (1).
q. Probability of long term repair - real [0, 1]. The sum of this probability and the probability of short term repair must not exceed one (1).
r. Rearm and refuel time - real in minutes.
s. Short term mean time to repair - real in hours. Short term service time is exponentially distributed with this mean service time.
t. Long term mean time to repair - real in hours. Long term service time is exponentially distributed with this mean service time.
u. Jammer probabilities. For each mission listed below, enter the conditional probability the aircraft's jammers are turned on given the aircraft is configured to carry jammers. For any mission the aircraft is not capable of flying enter a zero (0).
   (1) Close air support (CAS)
   (2) Battlefield interdiction (BAI)
   (3) Offensive counter-air (OCA)
   (4) Attack logistic facility
   (5) Attack C^3 facility
   (6) Attack supply train
   (7) Attack choke point
   (8) Attack transshipment point
   (9) Anti-ship
   (10) Strategic target interdiction (STI)
   (11) Suppress enemy air defense (SEAD)
   (12) Orbiting counter-air
   (13) Escorting counter-air
   (14) Reconnaissance/early warning (RECCE)
   (15) Resupply/reinforcement
   (16) Reserve
   (17) Move to dispersal base
v. Number of configurations - integer >= 1
w. Configuration information. For each configuration, enter the following:
   (1) Configuration name - 1 to 10 characters. Do not embed blanks.
   (2) Number of air munitions sticks - integer >= 0.
   (3) Stick information. If the number of sticks is greater than 0, enter the following for each stick:
      (a) Stick name - 1 to 10 characters. The name of a previously defined air munitions stick.
      (b) Number carried - integer.

(c) Circular error probability (CEP) - real. If the stick represents an air-to-ground munitions the CEP is the radius (meters) of the circle within which 50 percent of the munitions will land. If the stick represent an air-air munitions, enter a zero (0).

(4) Number of jammers - integer. Although the FTLM program data structure can accommodate multiple jammer types to be carried in a configuration, the current ground-to-air assessment algorithm assumes only one type of jammer is carried. Therefore, enter either a zero (0) or a one (1).

(5) Jammer information. If the number of jammers is 1, enter the following:

(a) Jammer name - 1 to 10 characters. The name of a previously defined jammer.

(b) Number carried - integer.

(6) Number of sensors - integer >= 0. This is a departure from TAC Thunder. TAC Thunder, allows at most only one type of sensor to be carried.

(7) Sensor information. If the number of sensors is greater than 0, enter the sensor name - 1 to 10 characters. The name of a previously defined sensor.

Aircraft data delimiter - "*"

20. Air Defense/Fire Support Type Records

TAC Thunder classifies air defenses into two types: PRIMARY and SECONDARY. The current FTLM ground-to-air algorithm limits assessments between a package and a PRIMARY air defense site. As a result, this section of the data base considers only PRIMARY air defenses. When SECONDARY sites are added to the algorithm, this section will incorporate SECONDARY air defense type data. For each air defense/fire support type, enter the following:

a. Type name - 1 to 10 characters. Do not embed blanks.

b. Launcher's equipment type name - 1 to 10 characters. The name of a previously defined equipment type.

c. Number of launchers - integer > 0.

d. Standard deviation of launchers - real > 0.

c. Number of ready rounds per launcher - integer.

e. System refire time - real in seconds.

f. Ground-air weapon name - 1 to 10 characters. If the system has an air defense capability, enter the name of a previously defined ground-air weapon. If no such capability exists, enter "NONE".

g. Air defense information. If the system has an air defense capability, enter the following:

(1) Probability a single missile is available - real.

(2) Maximum number of rounds stored - integer.

(3) Reorder level - real in the range [0.0, 1.0). The level of rounds, expressed as a fraction of the maximum number of rounds stored, at which a replenishment order is placed.

(4) Mean supply time - real in days. Mean time to wait for resupply after a replenishment order is placed. The waiting time is exponentially distributed with this mean time.

(5) Acquisition radar name - 1 to 10 characters. The name of a previously defined radar.

(6) Number of acquisition radar - integer.

(7) Fire control radar name - 1 to 10 characters. The name of a previously defined radar. If the acquisition radar also serves as the fire control radar, enter its name here, as well.

(8) Number of fire control radar - integer.

(9) Number of aircraft - integer. Number of aircraft that are vulnerable to the air defense type.

(10) Ground-to-air pK information. For each aircraft, enter the following:

(a) Aircraft name - 1 to 10 characters. The name of a previously defined aircraft.

(b) Probability of kill - real. If the pK is zero, the aircraft and its pK do not have to be listed. The FTLM program assumes that the air defense type is ineffective against any unlisted aircraft.

h. Ground munitions name - 1 to 10 characters. If the system has a fire support capability, enter the name of a previously defined ground munitions. If no such capability exists, enter "NONE".

i. Fire support information. If the system has a fire support capability, enter the following:

(1) Number of volleys fired - integer >= 1.

(2) Maximum number of rounds stored - integer.

(3) No fire level - real in the range [0.0, 1.0). The level of rounds, expressed as a fraction of the maximum number of rounds stored, below which a unit is not allowed to use. For example, if the no fire level is 0.5, a unit is not allowed to use more than 50% of its allocated rounds.

(4) Reorder level - real in the range [0.0, 1.0). The level of rounds, expressed as a fraction of the maximum number of rounds stored, at which a replenishment order is placed.

(5) Mean supply time - real in days. Mean time to wait for resupply after a replenishment order is placed. The waiting time is exponentially distributed with this mean time.

(6) Number of volleys - integer >= 1.

(7) Volley information. Enter the following for each volley:

(a) Volley name - 1 to 10 characters. The name of a previously defined volley.

(b) Circular error probability (CEP) - real.

Air defense/fire support type data delimiter - "*"

21. Air-Ground pK Records

Each probability of kill, $AGPK_{IJKL}$, is the result of air munitions stick I delivered by aircraft J against target component L of target type K. For each air-to-ground air munitions stick, enter the following:

a. Stick name - The name of a previously defined air munitions stick belonging to this side.
b. Number of aircraft - integer. Number of aircraft that can be configured to carry this stick.
c. pK information. Enter the following for each delivery aircraft:
   (1) Aircraft name - 1 to 10 characters. The name of a previously defined aircraft.
   (2) Combat unit component pKs. Enter the pK of this aircraft and stick against each equipment listed in the Equipment Type Records. These pKs must be listed in the same order as the equipment types appear in the Equipment Type Records.
   (3) Air base component pKs. Enter the pK of this aircraft and stick against each component listed below.
      (a) Runways
      (b) Aircraft
      (c) Maintenance facilities
      (d) Aircraft shelters
      (e) Transshipment facilities
      (f) Air munitions
      (g) Spare parts
      (h) POL
   (4) Logistic facility component pKs. Enter the pK of this aircraft and stick against each component listed below.
      (a) Issue capacity
      (b) Supplies
   (5) C^3 facility component pKs. Enter the pK of this aircraft and stick against each component listed below.
      (a) Antennas
      (b) Vans
   (6) Supply train pK. Enter the pK of this aircraft and stick against supplies.
   (7) Choke point pK. Enter the pK of this aircraft and stick against choke points.

(8) Transshipment point pK. Enter the pK of this aircraft and stick against transshipment point capacity.
(9) Strategic target pK Enter the pK of this aircraft and stick against a strategic target.
(10) Air defense unit component pKs. Enter the pK of this aircraft and stick against the component listed below
(a) Radar
(b) Launchers

Air-ground pK data delimiter - "*"

22. Ground-Ground pK Records

For each ground-to-ground volley, enter the following:

a. Volley name - The name of a previously defined volley.
b. Number of ground munitions - integer. This number should always be one (1).
c. pK information. Enter the following for each ground munitions:
(1) Ground munitions name - 1 to 10 characters. The name of the ground munitions that the volley is derived from.
(2) Combat unit component pKs. Enter the pK of this volley against each equipment listed in the Equipment Type Records. These pKs must be listed in the same order as the equipment types appear in the Equipment Type Records.
(3) Air base component pKs. Enter the pK of this volley against each component listed below.
(a) Runways
(b) Aircraft
(c) Maintenance facilities
(d) Aircraft shelters
(e) Transshipment facilities
(f) Air munitions
(g) Spare parts
(h) POL
(4) Logistic facility component pKs. Enter the pK of this volley against each component listed below.
(a) Issue capacity
(b) Supplies
(5) C^3 facility component pKs. Enter the pK of this volley against each component listed below.
(a) Antennas
(b) Vans
(6) Supply train pK. Enter the pK of this volley against supplies.
(7) Choke point pK. Enter the pK of this volley against choke points.

(8) Transshipment point pK. Enter the pK of this volley against transshipment point capacity
(9) Strategic target pK. Enter the pK of this volley against the strategic targets.
(10) Air defense unit component pKs. Enter the pK of this volley against the component listed below.
(a) Radar
(b) Launchers

Ground-ground pK data delimiter - "*"

23. Atom Type Records

An atom is the smallest ground unit that will be represented in the scenario. The atom serves two purposes. First, it serves as the basis for splitting combat units. Second, it serves as the basic unit that the Type I sensors track. For each atom type, enter the following:

a. Faction name - 1 to 10 characters. The name of a previously defined faction.
b. Type - character. Enter one of the following:
INFANTRY
ARMOR
MECHANIZED
CAVALRY
AIRBORNE
ARTILLERY
c. Movement restriction
1 = Tracked vehicles
2 = Wheeled vehicles
3 = Foot
4 = Heavy equipment (primary roads only)
d. Ammunition consumption when not in combat - real. Tons per day.
e. Ammunition consumption in offensive operations - real. Tons per day.
f. Ammunition consumption in defensive operations - real. Tons per day.
g. POL consumption when not in combat - real. Gallons per day.
h. POL consumption in offensive operations - real. Gallons per day.
i. POL consumption in defensive operations - real. Gallons per day.
j. Other supply consumption when not in combat - real. Tons per day.
k. Other supply consumption in offensive operations - real. Tons per day.
l. Other supply consumption in defensive operations - real. Tons per day.
m. Flat terrain speed - real in kilometers per hour.
n. Rolling terrain speed - real in kilometers per hour
o Severe terrain speed - real in kilometers per hour.

p. Number of equipment types owned by the atom - integer.
q. Equipment information. For each equipment type the atom is authorized to own, enter the following:
   (1) Equipment name - 1 to 10 characters. The name of a previously defined equipment type.
   (2) Initial amount - integer > 0.
   (3) Standard deviation - real > 0.

If this atom will be supported by a primary air defense type and/or one or more fire support types, *do not* include the equipment types listed with those air defense/fire support types, otherwise, these equipment types will be counted twice.

r. Primary air defense type name - 1 to 10 characters. The name of a previously defined primary air defense type. If none, enter "NONE". Only one primary air defense type is allowed per atom.
s. Number of fire support types - integer >= 0.
t. Fire support type list. For each fire support type supporting the atom, enter the fire support type name - 1 to 10 characters. The name of a previously defined fire support type.

Atom type data delimiter - "*"

24. Combat Unit Records

For each combat unit, enter the following:

a. Faction name - 1 to 10 characters. The name of a previously defined faction.
b. Unit name - 1 to 10 characters. Do not embed blanks.
c. Type - character. Enter one of the following:
   INFANTRY
   ARMOR
   MECHANIZED
   CAVALRY
   AIRBORNE
   ARTILLERY
d. Size - character. Enter one of the following:
   PLATOON
   COMPANY
   BATTALION
   REGIMENT
e. Headquarters - 1 to 10 characters. The name of a previously defined unit. If none, enter "NONE".
f. Reserved - real.
g. Reserved - real.
h. Combat threshold - real. Level of combat strength at which the unit breaks. Entered as a fraction in the range [0.0, 1.0]

i. Logistics threshold - real. Level of logistics strength at which the unit breaks. Entered as a fraction in the range [0.0, 1.0]
j. Number of atoms - integer > 0. If the unit's size is equal to the atom size this number should be one (1), otherwise, this number should be greater than 1.
k. Atom list. List each atom by type. There are currently six types of atoms recognized by the model:
INFANTRY
ARMOR
MECHANIZED
CAVALRY
AIRBORNE
ARTILLERY

For example, if the unit is composed of three armored and two mechanized atoms, the atoms may be listed as ARMOR ARMOR ARMOR MECHANIZED MECHANIZED. The order that the names appear does not matter. The atoms could just as well have been listed as MECHANIZED MECHANIZED ARMOR ARMOR ARMOR.

l. Unit radius. Enter the unit's radius (meters) for each posture listed below.
(1) Stationary
(2) Moving
(3) Obstacle delay
(4) Meeting engagement
(5) Attack
(6) Deliberate defense
(7) Hasty defense
(8) Ambush

Combat unit data delimiter - "*"

25. Air Base Records

For each air base, enter the following:

a. Faction name - 1 to 10 characters. The name of a previously defined faction.
b. Air base name - 1 to 10 characters. Do not embed blanks.
c. Headquarters - 1 to 10 characters. The name of a previously defined unit. If none, enter "NONE".
d. Location - 1 to 10 characters. The name of a physical node used as an air base (use = 1). This is the physical node where the air base starts the scenario.
e. Time arrives at location - real in decimal days. This is the time that the air base enters the scenario. For example, if the time is 1.25, the unit arrives at its entry point at 0600 of the second day of the simulation.
f. Primary air defense type name - 1 to 10 characters. The name of a previously defined primary air defense type. If none, enter "NONE". Only one primary air defense type is allowed per air base.

g. Air base radius - real in meters.
h. Component radius. Enter the radius (meters) for each component listed below.
   (1) Maintenance facility
   (2) Air munitions
   (3) Spares
   (4) POL
   (5) Transshipment supplies
   (6) Aircraft in the open
   (7) Aircraft shelters
i. Soft/hard target flag. Enter either a 1 (soft target) or a 2 (hard target) for each component listed below.
   (1) Maintenance facility
   (2) Air munitions
   (3) Spares
   (4) POL
j. Reserved - integer. Enter a zero (0).
k. Reserved - integer. Enter a zero (0).

Air base data delimiter - "*"

26. Squadron Records

For each squadron, enter the following:
a. Faction name - 1 to 10 characters. The name of a previously defined faction.
b. Squadron name - 1 to 10 characters. Do not embed blanks.
c. Headquarters - 1 to 10 characters. The name of a previously defined unit. If none, enter "NONE".
d. Main operating base - 1 to 10 characters. The name of a previously defined air base unit or vessel.
e. Time arrives at main operating base - real in decimal days. This is the time that the squadron enters the scenario.
f. Aircraft name - 1 to 10 characters. The name of the aircraft flown by the squadron.
h. Squadron effectiveness. For each mission listed below, enter the squadron's relative effectiveness number. This number is an integer in the range [0, 100]. For any mission the squadron is not capable of flying enter a zero (0).
   (1) CAS
   (2) BAI
   (3) OCA
   (4) Attack logistic facility
   (5) Attack C^3 facility
   (6) Attack supply train
   (7) Attack choke point
   (8) Attack transshipment point

(9) Anti-ship
(10) STI
(11) SEAD
(12) Orbiting counter-air
(13) Escorting counter-air
(14) RECCE
(15) Resupply/reinforcement
(16) Reserve
(17) Move to dispersal base

Squadron data delimiter - "*"

27. Corridor Records

Corridor records are used in conjunction with Course of Action records to restrict the movement of combat units. The planning module attempts to find an optimal path to the major objective within the corridor. For each corridor, enter the following:

a. Corridor name - 1 to 10 characters. Do not embed blanks.
b. Number of physical nodes - integer.
c. Physical node information. For each physical node in the corridor, enter the physical node name - 1 to 10 characters.
d. Number of transit nodes - integer.
e. Transit node information. For each transit node in the corridor, enter the transit node name - 1 to 10 characters.

Corridor data delimiter - "*"

28. Course of Action Records

For each course of action, enter the following:

a. Side name - 1 to 10 characters. The name of a previously defined side.
b. Course of action name - 1 to 10 characters. Do not embed blanks.
c. Utilization flag. If the side represents the attacker, this flag indicates whether the course of action will be followed by the side. For the attacking side only one course of action may have this flag set to 1; the remaining courses of action must have this flag set to 0. If the side represents the defender, set the flag to 0.
d. Link ID - integer. "Links" the defender's course of action to an attacker's course of action. For example, suppose two sides, RED and BLUE, have been defined and RED has been chosen to be the attacker. Suppose RED has three courses of action defined called R.COA.1, R.COA.2, and R.COA.3. BLUE must have three courses of action defined, one assigned to counter each RED course of action. Suppose these courses of action are called B.COA.1 (to

counter R.COA.1), B.COA.2 (to counter R.COA.2), and B.COA.3 (to counter R.COA.3). The link ID allows the program to identify each RED course of action with its BLUE counterpart. Each course of action/counter-course of action pair is assigned an ID number as shown below. Both R.COA.1 and B COA.1 are assigned link ID number 1 Likewise, link ID number 2 is assigned to both R.COA.2 and B.COA.2, and so on.

Courses of Action

RED	BLUE	Link ID
R.COA.1	B.COA.1	1
R.COA.2	B.COA.2	2
R.COA.3	B.COA.3	3

e. Number of combat units - integer.

f. Assigned units. For each combat unit, enter the following:

(1) Unit name - 1 to 10 characters. The name of a previously defined combat unit.

(2) Corridor name - 1 to 10 characters. The corridor the unit will be restricted to follow.

(3) Unit link ID - integer. "Links" the unit across courses of action.

(4) Major objective - 1 to 10 characters. The name of a previously defined physical node. This node must exist in the corridor assigned to the unit.

(5) Reserved - character. Enter NONE.

(6) Initial immediate objective - 1 to 10 characters. The name of a previously defined physical node. This is the physical node where the unit starts the scenario. This node must exist in the corridor assigned to the unit.

(7) Reserve flag.

0 = unit is not used as a reserve

1 = unit is used as a reserve

(8) Time arrives at initial immediate objective - real in decimal days. This is the time that the unit enters the scenario.

Course of action data delimiter - "*"

APPENDIX B. STLM INITIALIZATION DATA

The following is a sample initialization file for the STLM program. The file is entered as a plain text ASCII text file with a NET extension. Appendix A outlines the requirements for data entry.

```
2   195905   1.0   0   1.0   1.0   50.0   25.0   5.0   0.001   0.5   0.5   1
* end of parameter data - start side data

BLUE
1    1    1    1    1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
1.0  1.0  0.1  0.1  0.1  1.0  1.0  1.0  1.0  0.5  0.5  0.5  0    1    1.0  1.0  0.5
RED
2    1    1    1    1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0
1.0  1.0  0.1  0.1  0.1  1.0  1.0  1.0  1.0  0.5  0.5  0.5  0    1    1.0  1.0  0.5
* end of side data - start side relationship data

BLUE   RED   2
* end of side relationship data - start faction data

RIGHT   BLUE   PLATOON   COMPANY   LEFT   RED   COMPANY   BATTALION
* end of faction data - start physical node data

Node.01   1   2.   12.   3.5   5   1   7.0   0   0.   0   0   0   0.   0.   0.   0.   0
Node.02   2   2.    8.   3.0   5   1   6.0   0   0.   0   0   0   0.   0.   0.   0.   0
Node.03   3   11.   6.   1.5   5   1   3.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.04   4   14.   8.   1.5   5   1   3.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.05   5   17.  11.   1.5   5   1   3.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.06   6   17.   5.   1.5   5   1   3.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.07   7   21.   8.   1.5   5   1   3.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.08   8   23.  19.   2.5   5   1   3.0   0   0.   0   0   0   0.   0.   0.   0.   0
Node.09   9   27.   5.   1.5   5   1   3.0   0   0.   0   0   0   0.   0.   0.   0.   0
Node.10  10   30.  19.   2.0   5   1   3.0   0   0.   0   0   0   0.   0.   0.   0.   0
Node.11  11   32.  13.   1.5   5   1   3.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.12  12   32.  10    1.5   5   1   3.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.13  13   33.  18.   1.0   5   1   3.0   0   0.   1   1   0   0.   0.   0.   0.   0
Node.14  14   35.  17.   1.5   3   3   5.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.15  15   35.   5.   1.5   5   1   3.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.16  16   38.  20.   1.5   3   2   3.0   0   0.   0   0   0   0.   0.   0.   0.   0
Node.17  17   38.  17.   2.5   3   3   5.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.18  18   41.  15.   1.5   3   3   3.0   0   0.   0   1   0   0.   0.   0.   0.   0
Node.19  19   42.  20.   2.5   3   2   5.0   0   0.   0   0   0   0.   0.   0.   0.   0
Node.20  20   44.  16.   1.5   3   2   3.0   0   0.   0   0   0   0.   0.   0.   0.   0
Node.21  21   46.  22.   2.0   1   1   4.0   0   0.   0   1   0   0.   0.   0.   0.   0
```

* end of physical node data - start arc/transit node data

```
Node.01 Node.02 1     Transit.01
        4.0     2     1    0    0    0    0    0    1    2.5    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.01 Node.04 1     Transit.02
        12.6    2     2    0    0    0    0    0    1    1.0    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.02 Node.03 1     Transit.03
        9.2     2     2    0    0    0    0    0    1    1.5    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.03 Node.04 1     Transit.04
        3.6     2     2    0    0    0    0    0    1    2.0    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.03 Node.06 1     Transit.05
        6.1     2     1    0    0    0    0    0    1    1.5    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.04 Node.05 1     Transit.06
        4.2     2     1    0    0    0    0    0    1    3.0    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.04 Node.06 1     Transit.07
        4.2     3     2    0    0    0    0    0    1    0.5    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.05 Node.07 1     Transit.08
        5.0     2     2    0    0    0    0    0    1    2.0    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.05 Node.08 1     Transit.09
        10.0    2     1    0    0    0    0    0    1    2.5    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.05 Node.11 1     Transit.10
        15.1    2     2    0    0    0    0    0    1    1.5    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.06 Node.07 1     Transit.11
        5.0     2     1    0    0    0    0    0    1    2.5    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.06 Node.09 1     Transit.12
        10.0    3     2    0    0    0    0    0    1    2.5    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.07 Node.12 1     Transit.13
        11.1    3     2    0    0    0    0    0    1    2.0    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.08 Node.10 1     Transit.14
        7.0     3     2    0    0    0    0    0    1    1.0    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.09 Node.12 1     Transit.15
        7.1     3     1    0    0    0    0    0    1    3.0    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.09 Node.15 1     Transit.16
        8.0     3     1    0    0    0    0    0    1    3.0    0
        0.0     0     0    0 BLUE 1000 RED 1
Node.10 Node.13 1     Transit.17
```

3.2 3 3 0 0 0 0 0 1 0.5 0
0.0 0 0 0 BLUE 1000 RED 1
Node.11 Node.12 1 Transit.18
3.0 3 1 0 0 0 0 0 1 3.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.11 Node.14 1 Transit.19
5.0 3 2 0 0 0 0 0 1 2.5 0
0.0 0 0 0 BLUE 1000 RED 1
Node.11 Node.17 1 Transit.20
7.2 3 2 0 0 0 0 0 1 2.5 0
0.0 0 0 0 BLUE 1000 RED 1
Node.12 Node.15 1 Transit.21
5.8 2 2 0 0 0 0 0 1 2.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.12 Node.17 1 Transit.22
9.2 3 2 0 0 0 0 0 1 3.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.13 Node.14 1 Transit.23
2.2 3 2 0 0 0 0 0 1 0.5 0
0.0 0 0 0 BLUE 1000 RED 1
Node.14 Node.16 1 Transit.24
4.2 3 2 0 0 0 0 0 1 3.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.14 Node.17 1 Transit.25
3.0 3 2 0 0 0 0 0 1 3.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.14 Node.19 1 Transit.26
7.6 3 2 0 0 0 0 0 1 3.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.15 Node.17 1 Transit.27
12.4 3 2 0 0 0 0 0 1 2.5 0
0.0 0 0 0 BLUE 1000 RED 1
Node.15 Node.18 1 Transit.28
11.7 3 2 0 0 0 0 0 1 2.5 0
0.0 0 0 0 BLUE 1000 RED 1
Node.16 Node.19 1 Transit.29
4.0 3 2 0 0 0 0 0 1 3.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.17 Node.18 1 Transit.30
3.6 3 2 0 0 0 0 0 1 2.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.17 Node.19 1 Transit.31
5.0 3 2 0 0 0 0 0 1 3.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.18 Node.19 1 Transit.32
5.1 3 2 0 0 0 0 0 1 0.5 0
0.0 0 0 0 BLUE 1000 RED 1
Node.18 Node.20 1 Transit.33
3.2 3 2 0 0 0 0 0 1 1.0 0
0.0 0 0 0 BLUE 1000 RED 1
Node.19 Node.20 1 Transit.34

```
        4.5     3     1     0     0     0     0     0     1     1.5     0
        0.0     0     0     0 BLUE 1000 RED 1
Node.19 Node.21 1     Transit.35
        4.5     3     3     0     0     0     0     0     1     1.0     0
        0.9     0     0     0 BLUE 1000 RED 1
Node.20 Node.21 1     Transit.36
        6.3     3     3     0     0     0     0     0     1     1.0     0
        0.9     0     0     0 BLUE 1000 RED 1
* end of arc/transit node data - start observation point data

BLUE   OP1   Node.01   5   27.0   16.0   1   Transit.02
BLUE   OP2   Node.02   5   30.0   6.0    1   Transit.03
BLUE   OP3   Node.03   5   27.0   16.0   2   Transit.04   Transit.05
BLUE   OP4   Node.04   5   27.0   16.0   2   Transit.06   Transit.07
BLUE   OP5   Node.05   5   27.0   16.0   2   Transit.09   Transit.10
BLUE   OP6   Node.06   5   30.0   6.0    2   Transit.11   Transit.12
BLUE   OP7   Node.07   5   30.0   6.0    1   Transit.13
BLUE   OP8   Node.08   5   30.0   6.0    1   Transit.14
BLUE   OP9   Node.09   5   30.0   6.0    2   Transit.15   Transit.16
* end of observation point data - start equipment data

RED.TROOPS   1   0.05   2   1   0   0   0   0   0   1   0   0   0   0   0
RED.TANK     1   0.90   2   1   0   0   0   0   0   0   0   0   0   0   0
RED.BMP      1   0.50   2   1   0   0   0   0   0   0   0   0   0   0   0
RED.ARTY     1   0.45   2   1   0   0   0   0   0   0   0   0   0   0   0
RAD_LNCHR1   1   1.00   2   0   1   0   0   0   0   0   0   0   0   0   0
BLU.TROOPS   1   0.08   2   1   0   0   0   0   0   1   0   0   0   0   0
BLUE.TANK    1   1.00   2   1   0   0   0   0   0   0   0   0   0   0   0
BLUE.IFV     1   0.60   2   1   0   0   0   0   0   0   0   0   0   0   0
BLUE.ARTY    1   0.45   2   1   0   0   0   0   0   0   0   0   0   0   0
* end of equipment data - start combat system data

6
RED.TANK    RED.TANK    1     0.0    2000.0   1.00   5.5
RED.BMP     RED.BMP     1     0.0    2500.0   0.75   0.8
RED.ARTY    RED.ARTY    2   100.0   15000.0   0.00   2.5
BLUE.TANK   BLUE.TANK   1     0.0    3000.0   1.00   6.0
BLUE.IFV    BLUE.IFV    1     0.0    3750.0   0.75   0.8
BLUE.ARTY   BLUE.ARTY   2   100.0   15000.0   0.00   3.0
* end of combat system data - start firer-target data

RED.TANK     3
        BLUE.TANK    0.85        0.6
        BLUE.IFV     0.15        0.7
        BLUE.ARTY    0.00        0.7
RED.BMP      3
        BLUE.TANK    0.80        0.15
        BLUE.IFV     0.20        0.45
        BLUE.ARTY    0.00        0.45
RED.ARTY     3
        BLUE.TANK    0.15        0.000131
```

	BLUE.IFV	0.15	0.000524
	BLUE.ARTY	0.70	0.004712
BLUE.TANK 3			
	RED.TANK	0.90	0.8
	RED.BMP	0.10	0.9
	RED.ARTY	0.00	0.9
BLUE.IFV 3			
	RED.TANK	0.80	0.8
	RED.BMP	0.20	0.8
	RED.ARTY	0.00	0.8
BLUE.ARTY 3			
	RED.TANK	0.10	0.000131
	RED.BMP	0.10	0.002094
	RED.ARTY	0.80	0.004712

* end of firer-target data - start air munitions data
* end of air munitions data - start ground munitions data
* end of ground munitions data - start munitions stick/volley data
* end of munitions stick/volley data - start surface-air weapon data

B.SAM.1	2000	1000	4000	5000
B.SAM.2	2500	2000	5000	2000
R.SAM.1	2000	1000	4000	5000
R.SAM.2	12000	30000	60000	20000
R.SAM.3	15000	40000	80000	25000

* end of surface-air weapon data - start radar data

RED.TA.1	10000.	10000.	5	0	1	360.	10.	0.	120000.	10.	15.
RED.TA.2	30000.	5000.	3	0	0	360.	5.	0.	160000.	10.	15.
RED.FC.1	20000.	6000.	0	2	0	360.	20.	50000.	110000.	15.	20.
RED.FC.2	40000.	7500.	0	2	0	360.	15.	70000.	100000.	15.	20.
RED.FC.3	70000.	9000.	0	2	0	360.	10.	100000.	105000.	15.	20.
RED.AC.1	10000.	0.	0	0	1	90.	0.	35000.	100000.	0.	0.
BLUE.TA.1	9000.	8000.	5	0	1	360.	10.	0.	10000.	10.	15.
BLUE.FC.1	7000.	5000.	0	2	0	360.	15.	6000.	8000.	15.	20.
BLUE.AC.1	9000.	0.	0	0	1	90.	0.	10000.	15000.	0.	0.

* end of radar data - start Type I sensor data

BLUE	B.SENSOR.1	4	RED.TANK	RED.BMP	RED.ARTY	RED.TRUCK
21	Node.01		0.1	0.08	0.09	0.05
	Node.02		0.1	0.08	0.09	0.05
	Node.03		0.1	0.08	0.09	0.05
	Node.04		0.1	0.08	0.09	0.05
	Node.05		0.1	0.08	0.09	0.05
	Node.06		0.1	0.08	0.09	0.05
	Node.07		0.1	0.08	0.09	0.05
	Node.08		0.1	0.08	0.09	0.05
	Node.09		0.1	0.08	0.09	0.05
	Node.10		0.1	0.08	0.09	0.05
	Node.11		0.1	0.08	0.09	0.05
	Node.12		0.1	0.08	0.09	0.05
	Node.13		0.1	0.08	0.09	0.05

	Node.14	0.1	0.08	0.09	0.05
	Node.15	0.1	0.08	0.09	0.05
	Node.16	0.1	0.08	0.09	0.05
	Node.17	0.1	0.08	0.09	0.05
	Node.18	0.1	0.08	0.09	0.05
	Node.19	0.1	0.08	0.09	0.05
	Node.20	0.1	0.08	0.09	0.05
	Node.21	0.1	0.08	0.09	0.05
36	Transit.01	0.1	0.08	0.09	0.05
	Transit.02	0.1	0.08	0.09	0.05
	Transit.03	0.1	0.08	0.09	0.05
	Transit.04	0.1	0.08	0.09	0.05
	Transit.05	0.1	0.08	0.09	0.05
	Transit.06	0.1	0.08	0.09	0.05
	Transit.07	0.1	0.08	0.09	0.05
	Transit.08	0.1	0.08	0.09	0.05
	Transit.09	0.1	0.08	0.09	0.05
	Transit.10	0.1	0.08	0.09	0.05
	Transit.11	0.1	0.08	0.09	0.05
	Transit.12	0.1	0.08	0.09	0.05
	Transit.13	0.1	0.08	0.09	0.05
	Transit.14	0.1	0.08	0.09	0.05
	Transit.15	0.1	0.08	0.09	0.05
	Transit.16	0.1	0.08	0.09	0.05
	Transit.17	0.1	0.08	0.09	0.05
	Transit.18	0.1	0.08	0.09	0.05
	Transit.19	0.1	0.08	0.09	0.05
	Transit.20	0.1	0.08	0.09	0.05
	Transit.21	0.1	0.08	0.09	0.05
	Transit.22	0.1	0.08	0.09	0.05
	Transit.23	0.1	0.08	0.09	0.05
	Transit.24	0.1	0.08	0.09	0.05
	Transit.25	0.1	0.08	0.09	0.05
	Transit.26	0.1	0.08	0.09	0.05
	Transit.27	0.1	0.08	0.09	0.05
	Transit.28	0.1	0.08	0.09	0.05
	Transit.29	0.1	0.08	0.09	0.05
	Transit.30	0.1	0.08	0.09	0.05
	Transit.31	0.1	0.08	0.09	0.05
	Transit.32	0.1	0.08	0.09	0.05
	Transit.33	0.1	0.08	0.09	0.05
	Transit.34	0.1	0.08	0.09	0.05
	Transit.35	0.1	0.08	0.09	0.05
	Transit.36	0.1	0.08	0.09	0.05

* end of Type I sensor data - start jammer data
* end of jammer data - start altitude data

BLUE	75.	50.	1000.	500.	750.	900.
RED	75.	50.	1000.	500.	750.	900.

* end of altitude data - start aircraft data

```
BLUE.HELO
  0              0    2    NONE    1500.    1000.    2000.           1500.    1000.    700.
 11.             0.   0.      1       0.       0.      10.             2.0      8.0      0
  0              0    0       0       0        0        0               0        0       0
  0              0    0       0       0        0        1     B.CONFIG.1         0       0
  1    B.SENSOR.1
* end of aircraft data - start air defense/fire support type data

RED.ADA.1
RAD_LNCHR1          1        0.5       3     120.0
R.SAM.1          0.75         30    0.75         1
RED.TA.1            1
RED.FC.1            1          1
BLUE.HELO         0.5     NONE
* end of air defense/fire support type data - start air-surface pK data
* end of air-surface pK data - start surface-surface pK data
* end of surface-surface pK data - start atom data

RIGHT   ARMOR          1
        1.0      10.0      10.0     5.0     20.0     15.0     1.0     10.0     10.0     50.0     50.0
        50.0     2
        BLU.TROOPS     50       5.0
        BLUE.TANK      3        0.5
        NONE           0
RIGHT   MECHANIZED     1
        1.0      10.0      10.0     5.0     20.0     15.0     1.0     10.0     10.0     50.0     50.0
        50.0     2
        BLU.TROOPS     50       5.0
        BLUE.IFV       3        0.5
        NONE           0
RIGHT   ARTILLERY      1
        1.0      10.0      10.0     5.0     20.0     15.0     1.0     10.0     10.0     50.0     50.0
        50.0     2
        BLU.TROOPS     50       5.0
        BLUE.ARTY      4        0.5
        NONE           0
LEFT    ARMOR          1
        1.0      10.0      10.0     5.0     20.0     15.0     1.0     10.0     10.0     25.0     20.0
        15.0     2
        RED.TROOPS     12       2.0
        RED.TANK       13       0.5
        RED.ADA.1      0
LEFT    ARTILLERY      1
        1.0      10.0      10.0     5.0     20.0     15.0     1.0     10.0     10.0     23.0     18.0
        13.0     2
        RED.TROOPS     16       2.5
        RED.ARTY       8        0.75
        NONE           0
LEFT    MECHANIZED     1
        1.0      10.0      10.0     5.0     20.0     15.0     1.0     10.0     10.0     24.0     19.0
        14.0     2
```

```
        RED.TROOPS    70      7.0
        RED.BMP       15      1.0
        RED.ADA.1     0
* end of atom data - start combat unit data

RIGHT   BLUE.1   ARMOR   COMPANY   NONE   0.60   0.005   0.5   0.0   4
        ARMOR        ARMOR        MECHANIZED MECHANIZED
          1000   1000   1000   1000   1000   1000   1000   1000   1000
RIGHT  BLUE.2 ARMOR        COMPANY    NONE  0.55   0.004   0.5    0.0    4
        ARMOR        ARMOR        ARMOR        MECHANIZED
          1000   1000   1000   1000   1000   1000   1000   1000   1000
RIGHT  BLUE.3 MECHANIZED  COMPANY    NONE  0.50   0.003   0.5    0.0    4
        MECHANIZED  MECHANIZED  ARMOR        ARMOR
          1000   1000   1000   1000   1000   1000   1000   1000   1000
RIGHT  BLUE.4 MECHANIZED  COMPANY    NONE  0.50   0.003   0.5    0.0    4
        MECHANIZED  MECHANIZED  MECHANIZED  ARMOR
          1000   1000   1000   1000   1000   1000   1000   1000   1000
RIGHT  BLUE.5 ARTILLERY   COMPANY    NONE  0.50   0.003   0.5    0.0    2
        ARTILLERY    ARTILLERY
          1000   1000   1000   1000   1000   1000   1000   1000   1000
LEFT    RED.1  ARMOR       BATTALION  NONE  0.70   0.005   0.5    0.0    3
        ARMOR        ARMOR        ARMOR
          500    1000   1000   1000   1000   1000   1000   1000   1000
LEFT    RED.2  MECHANIZED BATTALION  NONE  0.70   0.005   0.5    0.0    3
        MECHANIZED  MECHANIZED  ARMOR
          500    1000   1000   1000   1000   1000   1000   1000   1000
LEFT    RED.3  MECHANIZED BATTALION  NONE  0.70   0.005   0.5    0.0    3
        MECHANIZED  MECHANIZED  ARMOR
          500    1000   1000   1000   1000   1000   1000   1000   1000
LEFT    RED.4  MECHANIZED BATTALION  NONE  0.70   0.005   0.5    0.0    3
        MECHANIZED  MECHANIZED  ARMOR
          500    1000   1000   1000   1000   1000   1000   1000   1000
LEFT    RED.5  ARTILLERY   BATTALION  NONE  0.70   0.005   0.5    0.0    3
        ARTILLERY    ARTILLERY    ARTILLERY
          500    1000   1000   1000   1000   1000   1000   1000   1000
* end of combat unit data - start air base data

RIGHT     BLUE.BASE          NONE     Node.21    0.0        NONE
     600.       100.       100.       50.       50.       10.      100.       200.
       2          1          1          1         0         0
* end of air base data - start squadron data

RIGHT  BLUE.AIR.1    NONE   BLUE.BASE    0.0      BLUE.HELO
     80        50        0        50        80        50        50        0        0
      0        50        0         0       100         0       100      100
* end of squadron data - start corridor data

NORTH
12            Node.01      Node.02      Node.03      Node.04      Node.05      Node.08
              Node.10      Node.13      Node.14      Node.16      Node.19      Node.21
13            Transit.01   Transit.02   Transit.03   Transit.04   Transit.06   Transit.09
```

	Transit.14	Transit.17	Transit.23	Transit.24	Transit.26	Transit.29
	Transit.35					
NO_CENTRAL						
10	Node.01	Node.02	Node.03	Node.04	Node.05	Node.11
	Node.14	Node.17	Node.19	Node.21		
12	Transit.01	Transit.02	Transit.03	Transit.04	Transit.06	Transit.10
	Transit.19	Transit.20	Transit.25	Transit.26	Transit.31	Transit.35
SO_CENTRAL						
11	Node.01	Node.02	Node.03	Node.04	Node.05	Node.06
	Node.07	Node.12	Node.17	Node.19	Node.21	
13	Transit.01	Transit.02	Transit.03	Transit.04	Transit.05	Transit.06
	Transit.07	Transit.08	Transit.11	Transit.13	Transit.22	Transit.31
	Transit.35					
SOUTH						
12	Node.01	Node.02	Node.03	Node.04	Node.06	Node.09
	Node.15	Node.17	Node.18	Node.19	Node.20	Node.21
17	Transit.01	Transit.02	Transit.03	Transit.04	Transit.05	Transit.07
	Transit.12	Transit.16	Transit.27	Transit.28	Transit.30	Transit.31
	Transit.32	Transit.33	Transit.34	Transit.35	Transit.36	

* end of corridor data - start course of action data

BLUE B.COA.1	0	1	5				
	BLUE.1	NORTH	1 Node.14	NONE	Node.14	0	0.0
	BLUE.2	SO_CENTRAL	2 Node.17	NONE	Node.17	0	0.0
	BLUE.3	SOUTH	3 Node.18	NONE	Node.18	0	0.0
	BLUE.4	NORTH	4 Node.14	NONE	Node.19	1	0.0
	BLUE.5	NORTH	5 Node.19	NONE	Node.19	0	0.0
BLUE B.COA.2	0	2	5				
	BLUE.1	NORTH	1 Node.14	NONE	Node.14	0	0.0
	BLUE.2	NO_CENTRAL	2 Node.17	NONE	Node.17	0	0.0
	BLUE.3	SOUTH	3 Node.18	NONE	Node.18	0	0.0
	BLUE.4	NO_CENTRAL	4 Node.17	NONE	Node.19	1	0.0
	BLUE.5	NO_CENTRAL	5 Node.19	NONE	Node.19	0	0.0
BLUE B.COA.3	0	3	5				
	BLUE.1	NORTH	1 Node.14	NONE	Node.14	0	0.0
	BLUE.2	SO_CENTRAL	2 Node.17	NONE	Node.17	0	0.0
	BLUE.3	SOUTH	3 Node.18	NONE	Node.18	0	0.0
	BLUE.4	SOUTH	4 Node.18	NONE	Node.19	1	0.0
	BLUE.5	SO_CENTRAL	5 Node.19	NONE	Node.19	0	0.0
RED	R.COA.1	1	1	5			
	RED.1	NORTH	1 Node.19	NONE	Node.01	0	0.0
	RED.2	NORTH	2 Node.19	NONE	Node.02	0	0.0
	RED.3	NORTH	3 Node.19	NONE	Node.02	0	0.0
	RED.4	NORTH	4 Node.19	NONE	Node.02	0	0.0
	RED.5	NORTH	5 Node.16	NONE	Node.01	0	0.0
RED	R.COA.2	0	2	5			
	RED.1	NO_CENTRAL	1 Node.19	NONE	Node.01	0	0.0
	RED.2	SOUTH	2 Node.19	NONE	Node.02	0	0.0

	RED.3	SOUTH	3 Node.19	NONE	Node.02	0	0.0
	RED.4	SO_CENTRAL	4 Node.19	NONE	Node.02	0	0.0
	RED.5	SO_CENTRAL	5 Node.17	NONE	Node.01	0	0.0
RED	R.COA.3	0	3	5			
	RED.1	SO_CENTRAL	1 Node.19	NONE	Node.01	0	0.0
	RED.2	NO_CENTRAL	2 Node.19	NONE	Node.02	0	0.0
	RED.3	SOUTH	3 Node.19	NONE	Node.02	0	0.0
	RED.4	SOUTH	4 Node.19	NONE	Node.02	0	0.0
	RED.5	SO_CENTRAL	5 Node.17	NONE	Node.01	0	0.0

* end of course of action data

APPENDIX C. NODE LISTING FOR NTC NETWORK

A. PHYSICAL NODES

The following table lists the physical nodes based on the terrain evaluation used in the STLM scenario. The X and Y grid are arbitrary and are not associated with military grid coordinates. The attributes listed are for reference only and do not influence the operation of the model.

Node	X Grid	Y Grid	Attribute
1	07	12	Red Assembly Area, Road Junction
2	07	08	Red Assembly Area, Road Junction
3	11	06	Road Junction
4	14	08	Road Junction
5	17	11	Road Junction
6	17	05	Road Junction
7	21	08	Road Junction
8	23	19	Change in Terrain
9	27	05	Road Junction
10	30	19	Change in Terrain
11	32	13	Road Junction
12	32	10	Road Junction
13	33	18	Change in Terrain
14	35	17	Primary Defensive Position
15	35	05	Road Junction
16	38	20	Secondary Defensive Position
17	38	17	Primary Defensive Position
18	41	15	Primary Defensive Position
19	42	20	Secondary Defensive Position
20	44	16	Secondary Defensive Position
21	46	22	Defensive Position/Blue Air Base

B. TRANSIENT NODES

The following is the list of transient nodes used in the NTC network. The defined arcs are connected by two physical nodes. The distance and width are measured in kilometers. The type of terrain impacts the movement rate of assets. Movement rates are defined in the initialization files, Appendix A.

Arc	Distance	Width	Terrain
1-2	4.0	2.5	Flat
1-4	8.1	1.0	Rolling
2-3	4.5	1.5	Rolling
3-4	3.6	2.0	Rolling
3-6	6.1	1.5	Flat
4-5	4.2	3.0	Flat
4-6	4.2	0.5	Rolling
5-7	5.0	2.0	Rolling
5-8	10.0	2.5	Flat
5-11	15.1	1.5	Rolling
6-7	5.0	2.5	Flat
6-9	10.0	2.5	Rolling
7-12	11.1	2.0	Rolling
8-10	7.0	1.0	Rolling
9-12	7.1	3.0	Flat
9-15	8.0	3.0	Flat
10-13	3.2	0.5	Severe
11-12	3.0	3.0	Flat
11-14	5.0	2.5	Rolling
11-17	7.2	2.5	Rolling
12-15	5.8	2.0	Rolling
12-17	9.2	3.0	Rolling
13-14	2.2	1.0	Rolling
14-16	4.2	3.0	Rolling
14-17	3.0	3.0	Rolling
14-19	7.6	3.0	Rolling
15-17	12.4	2.5	Rolling
15-18	11.7	2.5	Rolling
16-19	4.0	3.0	Rolling
17-18	3.6	2.0	Rolling
17-19	5.0	3.0	Rolling
18-19	5.1	0.5	Rolling
18-20	3.2	1.0	Rolling
19-20	4.5	1.5	Flat
19-21	4.5	1.5	Rolling
20-21	6.3	6.3	Severe

APPENDIX D. OUTPUT OF SENSOR SAMPLES

The following table is a collection of three samples of sensor observations for three different sensors used in STLM. The table lists the sensor used, the unit combinations that comprise 90% of the posterior probability (P(Unit)), the expected number of assets given that unit combination, and the observed sensor assets count. The observed count is aligned with the actual unit combination, designated by a double asterisk in the units column. The order of units in the units column is armor battalions, artillery battalions, and mechanized battalions.

			Expected			Observed		
Sensor	**Units***	**P(Unit)**	**Tank**	**Arty**	**BMP**	**Tank**	**Arty**	**BMP**
[redacted]								
	(3,0,6)**	1.00	39	0	93	39	0	90
	(1,3,2)**	1.00	14	22	30	14	22	30
	(2,0,4)**	1.00	24	0	61	24	0	61
[redacted]								
	(0,1,0)	0.17	0	8	0			
	(0,0,0)**	0.79	0	0	0	0	0	0
	(0,0,0)	0.30	0	0	0			
	(0,1,0)**	0.65	0	8	0	0	6	0
	(3,0,5)	0.06	39	0	78			
	(2,0,6)	0.07	26	0	89			
	(3,1,6)	0.15	39	8	89			
	(3,0,6)**	0.67	39	0	89	37	0	86
[redacted]								
	(0,3,1)	0.01	0	24	15			
	(2,1,1)	0.01	26	8	15			
	(2,2,0)	0.01	26	16	0			
	(2,0,1)	0.01	26	0	15			
	(1,3,0)	0.02	13	24	0			
	(1,2,1)	0.02	13	16	15			
	(0,3,0)	0.02	0	24	0			
	(2,1,0)	0.02	26	8	0			

			Expected			Observed		
Sensor	Units*	P(Unit)	Tank	Arty	BMP	Tank	Arty	BMP
	(2,0,0)	0.03	26	0	0			
	(0,2,1)	0.03	0	16	15			
	(1,1,1)	0.04	13	8	15			
	(1,2,0)	0.04	13	16	0			
	(1,0,1)	0.05	13	0	15			
	(0,1,1)	0.06	0	8	15			
	(0,2,0)	0.06	0	16	0			
	(0,0,1)	0.07	0	0	15			
	(1,1,0)	0.07	13	8	0			
	(1,0,0)	0.09	13	0	0			
	(0,1,0)	0.11	0	0	0			
	(0,0,0)**	0.13	0	0	0	0	0	0
	(1,2,1)	0.02	13	16	15			
	(0,1,1)	0.02	0	8	15			
	(2,1,0)	0.02	26	8	0			
	(1,1,1)	0.03	13	8	15			
	(0,3,0)**	0.03	0	24	0	0	17	0
	(1,3,0)	0.04	13	24	0			
	(0,0,0)	0.07	0	0	0			
	(1,0,0)	0.08	13	0	0			
	(0,2,0)	0.12	0	16	0			
	(1,2,0)	0.14	13	16	0			
	(0,1,0)	0.16	0	8	0			
	(1,1,0)	0.18	13	8	0			
	(5,1,6)	0.05	65	8	92			
	(5,0,6)	0.06	65	0	92			
	(4,2,6)	0.06	52	16	92			
	(3,1,6)	0.11	39	8	92			
	(3,0,6)**	0.13	39	0	92	61	7	110
	(4,1,6)	0.23	52	8	92			
	(4,0,6)	0.28	52	0	92			

*Unit combinations are (Armor, Artillery, Mechanized)

**Actual unit combination

LIST OF REFERENCES

1. Youngren, M.A., "FTLM Summary of Model Concept," CFAD, J-8, The Joint Staff, August 1993.

2. Department of the Army, "The Tools: Models and Simulations," Louisiana Maneuvers Task Force, Fort Monroe, VA, 27 May 1993.

3. Schmidt, Karl Morris, "Design Methodology for FTLM", Master's Thesis, Naval Postgraduate School, Monterey, CA, September 1993.

4. Johnson, Michael J., "Quantifying the Value of Reconnaissance Using Lanchestrian Type Equations", Master's Thesis, Naval Postgraduate School, Monterey, CA, March 1994.

5. Office of the Joint Chief of Staff, J-8, "Mission Need Statement-FTLM," Washington, DC, September 1993.

6. Department of the Army, Field Manual 100-5 "Operations", Headquarters, Department of the Army, Washington, DC, 14 June 1993.

7. Gaver, D.P., Jacobs, P.A., "Course-of-Action Realization and Perception in FTLM," Naval Postgraduate School, Monterey, CA.

8. Hartman, J.K., Parry, S.H., Caldwell, W.J., "Airland Combat Models II, Aggregate Combat Modeling", Naval Postgraduate School, Monterey, CA, September 1991.

9. Johnson, Michael J., and others, "Various Maximization Methods for Use with a Combat Simulation," Naval Postgraduate School, Monterey, CA, March 1994.

10. Yamauchi, Harold, "FTLM "C" Code and FTLM Scenario Data File Record," Roland's and Associates, Monterey, CA, July 1994.

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6154	2
2	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Commander and Director U.S. Army TRADOC Analysis Command TRAC-Monterey Monterey, CA 93943	1
4.	Commander U.S. Army TRADOC Analysis Command ATTN: ATRC (Mr. Bauman) Fort Leavenworth, KS 66027-5200	1
5.	Director U.S. Army TRADOC Analysis Command ATTN: ATRC-FOQ (Technical Information Center) Fort Leavenworth, KS 66027-5200	1
6.	Professor Samuel H. Parry Department of Operations Research Naval Postgraduate School, Code OR/Py Monterey, CA 93943	5
7.	Lieutenant Colonel George C. Prueitt Department of Operations Research Naval Postgraduate School, Code OR/Pt Monterey, CA 93943	1
8.	Captain Thomas A. Nelson 5725 Brown Ave Apt B Fort Knox, KY 40121	3